

METHOD FOR INTERPRETING AND EXECUTING USER

PREFERENCES OF AUDIOVISUAL INFORMATION

Lhs.
A₁
INSB 1

BACKGROUND OF THE INVENTION

5 The present invention relates to a system for managing audiovisual information, and in particular to a system for audiovisual information browsing, filtering, searching, archiving, and personalization.

10 Video cassette recorders (VCRs) may record video programs in response to pressing a record button or may be programmed to record video programs based on the time of day. However, the viewer must program the VCR based on information from a television guide to identify relevant programs to record. After recording, the viewer scans through the entire video tape to select relevant portions of the program for viewing using the functionality provided by the VCR, such as fast forward and fast reverse. Unfortunately, the searching and viewing is based on a linear search, which may require 15 significant time to locate the desired portions of the program(s) and fast forward to the desired portion of the tape. In addition, it is time consuming to program the VCR in light of the television guide to record desired programs. Also, unless the viewer recognizes the programs from the television guide as desirable it is unlikely that the viewer will select such programs to be recorded.

20 RePlayTV and TiVo have developed hard disk based systems that receive, record, and play television broadcasts in a manner similar to a VCR. The systems may be programmed with the viewer's viewing preferences. The systems use a telephone line

interface to receive scheduling information similar to that available from a television guide. Based upon the system programming and the scheduling information, the system automatically records programs that may be of potential interest to the viewer.

Unfortunately, viewing the recorded programs occurs in a linear manner and may require substantial time. In addition, each system must be programmed for an individual's preference, likely in a different manner.

Freeman et al., U.S. Patent No. 5,861,881, disclose an interactive computer system where subscribers can receive individualized content.

With all the aforementioned systems, each individual viewer is required to program the device according to his particular viewing preferences. Unfortunately, each different type of device has different capabilities and limitations which limit the selections of the viewer. In addition, each device includes a different interface which the viewer may be unfamiliar with. Further, if the operator's manual is inadvertently misplaced it may be difficult for the viewer to efficiently program the device.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

FIG. 1 is an exemplary embodiment of a program, a system, and a user, with associated description schemes, of an audiovisual system of the present invention.

FIG. 2 is an exemplary embodiment of the audiovisual system, including an analysis module, of FIG. 1.

FIG. 3 is an exemplary embodiment of the analysis module of FIG. 2.

FIG. 4 is an illustration of a thumbnail view (category) for the audiovisual system.

FIG. 5 is an illustration of a thumbnail view (channel) for the audiovisual system.

5 FIG. 6 is an illustration of a text view (channel) for the audiovisual system.

FIG. 7 is an illustration of a frame view for the audiovisual system.

FIG. 8 is an illustration of a shot view for the audiovisual system.

FIG. 9 is an illustration of a key frame view the audiovisual system.

10 FIG. 10 is an illustration of a highlight view for the audiovisual system.

FIG. 11 is an illustration of an event view for the audiovisual system.

FIG. 12 is an illustration of a character/object view for the audiovisual system.

15 FIG. 13 is an alternative embodiment of a program description scheme including a syntactic structure description scheme, a semantic structure description scheme, a visualization description scheme, and a meta information description scheme.

FIG. 14 is an exemplary embodiment of the visualization description scheme of FIG. 13.

20 FIG. 15 is an exemplary embodiment of the meta information description scheme of FIG. 13.

FIG. 16 is an exemplary embodiment of a segment description scheme for the syntactic structure description scheme of FIG. 13.

FIG. 17 is an exemplary embodiment of a region description scheme for the syntactic structure description scheme of FIG. 13.

FIG. 18 is an exemplary embodiment of a segment/region relation description scheme for the syntactic structure description scheme of FIG. 13.

5 FIG. 19 is an exemplary embodiment of an event description scheme for the semantic structure description scheme of FIG. 13.

FIG. 20 is an exemplary embodiment of an object description scheme for the semantic structure description scheme of FIG. 13.

FIG. 21 is an exemplary embodiment of an event/object relation graph description scheme for the syntactic structure description scheme of FIG. 13.

FIG. 22 is an exemplary embodiment of a user preference description scheme.

FIG. 23 is an exemplary embodiment of the interrelationship between a usage history description scheme, an agent, and the usage preference description scheme of FIG. 22.

FIG. 24 is an exemplary embodiment of the interrelationship between audio and/or video programs together with their descriptors, user identification, and the usage preference description scheme of FIG. 22.

FIG. 25 is an exemplary embodiment of a usage preference description scheme of FIG. 22.

FIG. 26 is an exemplary embodiment of the interrelationship between the usage description schemes and an MPEG-7 description schemes.

FIG. 27 is an exemplary embodiment of a usage history description scheme of FIG. 22.

FIG. 28 is an exemplary system incorporating the user history description scheme.

5 FIG. 29 is an exemplary user preferences description scheme.

FIG. 30 is an exemplary embodiment of a context diagram.

FIG. 31 is an exemplary embodiment of a filter agent.

FIG. 32 is an exemplary embodiment of a program description.

FIG. 33 is an exemplary embodiment of an individual preference.

10 FIG. 34 is an exemplary embodiment of a general user preference description.

FIG. 35 is an exemplary embodiment of a user preference description.

FIG. 36 is an exemplary embodiment of a mapping table.

FIG. 37 is an exemplary embodiment of a set of test operators.

15 FIG. 38 is an exemplary embodiment of combinatorial operators.

FIG. 39 is an exemplary embodiment of inter group combinatorial operator.

FIG. 40 is an exemplary embodiment of inter and intra group combinatorial operator.

20 FIG. 41 is an exemplary embodiment of inter/intra combinations.

FIG. 42 is an exemplary embodiment of inter/intra combinations supporting permutations.

FIG. 43 is an exemplary embodiment of constrained-AND combinatorial operator

FIGS. 44A and 44B are an exemplary embodiment of constrained-AND combinatorial operator.

FIGS. 45A and 45B are an exemplary embodiment of the operators.

FIGS. 46A and 46B are an exemplary embodiment of the operators.

FIGS. 47A-47C is an exemplary embodiment of is an example of a mapping table.

FIG. 48 is an exemplary embodiment of selected combinatorial operators.

FIG. 49 is an exemplary embodiment of cloning.

FIG. 50 is another exemplary embodiment of cloning.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

Many households today have many sources of audio and video information, such as multiple television sets, multiple VCR's, a home stereo, a home entertainment center, cable television, satellite television, internet broadcasts, world wide web, data services, specialized Internet services, portable radio devices, and a stereo in each of their vehicles. For each of these devices, a different interface is normally used to obtain, select, record, and play the video and/or audio content. For example, a VCR permits the selection of the recording times but the user has to correlate the television guide with the desired recording times. Another example is the user selecting a preferred set of preselected radio stations for his home stereo and also presumably selecting the

same set of preselected stations for each of the user's vehicles. If another household member desires a different set of preselected stereo selections, the programming of each audio device would need to be reprogrammed at substantial inconvenience.

5 The present inventors came to the realization that users of visual information and listeners to audio information, such as for example radio, audio tapes, video tapes, movies, and news, desire to be entertained and informed in more than merely one uniform manner. In other words, the audiovisual information presented to a particular user should be in a format and include content suited to their particular viewing preferences. In addition, the format should be dependent on the content of the particular
10 audiovisual information. The amount of information presented to a user or a listener should be limited to only the amount of detail desired by the particular user at the particular time. For example with the ever increasing demands on the user's time, the user may desire to watch only 10 minutes of or merely the highlights of a basketball game. In addition, the present inventors came to the realization that the necessity of
15 programming multiple audio and visual devices with their particular viewing preferences is a burdensome task, especially when presented with unfamiliar recording devices when traveling. When traveling, users desire to easily configure unfamiliar devices, such as audiovisual devices in a hotel room, with their viewing and listening preferences in a efficient manner.

20 The present inventors came to the further realization that a convenient technique of merely recording the desired audio and video information is not sufficient because the presentation of the information should be in a manner that is time efficient,

especially in light of the limited time frequently available for the presentation of such information. In addition, the user should be able to access only that portion of all of the available information that the user is interested in, while skipping the remainder of the information.

5 A user is not capable of watching or otherwise listening to the vast potential amount of information available through all, or even a small portion of, the sources of audio and video information. In addition, with the increasing information potentially available, the user is not likely even aware of the potential content of information that he may be interested in. In light of the vast amount of audio, image, and video information, the present inventors came to the realization that a system that records and presents to the user audio and video information based upon the user's prior viewing and listening habits, preferences, and personal characteristics, generally referred to as user information, is desirable. In addition, the system may present such information based on the capabilities of the system devices. This permits the system to record desirable information and to customize itself automatically to the user and/or listener. It is to be understood that user, viewer, and/or listener terms may be used interchangeability for any type of content. Also, the user information should be portable between and usable by different devices so that other devices may likewise be configured automatically to the particular user's preferences upon receiving the viewing information.

20 In light of the foregoing realizations and motivations, the present inventors analyzed a typical audio and video presentation environment to determine the significant portions of the typical audiovisual environment. First, referring to FIG. 1 the video,

image, and/or audio information 10 is provided or otherwise made available to a user and/or a (device) system. Second, the video, image, and/or audio information is presented to the user from the system 12 (device), such as a television set or a radio. Third, the user interacts both with the system (device) 12 to view the information 10 in a desirable manner and has preferences to define which audio, image, and/or video information is obtained in accordance with the user information 14. After the proper identification of the different major aspects of an audiovisual system the present inventors then realized that information is needed to describe the informational content of each portion of the audiovisual system 16.

With three portions of the audiovisual presentation system 16 identified, the functionality of each portion is identified together with its interrelationship to the other portions. To define the necessary interrelationships, a set of description schemes containing data describing each portion is defined. The description schemes include data that is auxiliary to the programs 10, the system 12, and the user 14, to store a set of information, ranging from human readable text to encoded data, that can be used in enabling browsing, filtering, searching, archiving, and personalization. By providing a separate description scheme describing the program(s) 10, the user 14, and the system 12, the three portions (program, user, and system) may be combined together to provide an interactivity not previously achievable. In addition, different programs 10, different users 14, and different systems 12 may be combined together in any combination, while still maintaining full compatibility and functionality. It is to be understood that the description scheme may contain the data itself or include links to the data, as desired.

A program description scheme 18 related to the video, still image, and/or audio information 10 preferably includes two sets of information, namely, program views and program profiles. The program views define logical structures of the frames of a video that define how the video frames are potentially to be viewed suitable for efficient browsing. For example the program views may contain a set of fields that contain data for the identification of key frames, segment definitions between shots, highlight definitions, video summary definitions, different lengths of highlights, thumbnail set of frames, individual shots or scenes, representative frame of the video, grouping of different events, and a close-up view. The program view descriptions may contain thumbnail, slide, key frame, highlights, and close-up views so that users can filter and search not only at the program level but also within a particular program. The description scheme also enables users to access information in varying detail amounts by supporting, for example, a key frame view as a part of a program view providing multiple levels of summary ranging from coarse to fine. The program profiles define distinctive characteristics of the content of the program, such as actors, stars, rating, director, release date, time stamps, keyword identification, trigger profile, still profile, event profile, character profile, object profile, color profile, texture profile, shape profile, motion profile, and categories. The program profiles are especially suitable to facilitate filtering and searching of the audio and video information. The description scheme enables users to have the provision of discovering interesting programs that they may be unaware of by providing a user description scheme. The user description scheme provides information to a software agent that in turn performs a search and filtering on behalf of the user by

possibly using the system description scheme and the program description scheme information. It is to be understood that in one of the embodiments of the invention merely the program description scheme is included.

Program views contained in the program description scheme are a feature that supports a functionality such as close-up view. In the close-up view, a certain image object, e.g., a famous basketball player such as Michael Jordan, can be viewed up close by playing back a close-up sequence that is separate from the original program. An alternative view can be incorporated in a straightforward manner. Character profile on the other hand may contain spatio-temporal position and size of a rectangular region around the character of interest. This region can be enlarged by the presentation engine, or the presentation engine may darken outside the region to focus the user's attention to the characters spanning a certain number of frames. Information within the program description scheme may contain data about the initial size or location of the region, movement of the region from one frame to another, and duration and terms of the number of frames featuring the region. The character profile also provides provision for including text annotation and audio annotation about the character as well as web page information, and any other suitable information. Such character profiles may include the audio annotation which is separate from and in addition to the associated audio track of the video.

The program description scheme may likewise contain similar information regarding audio (such as radio broadcasts) and images (such as analog or digital photographs or a frame of a video).

The user description scheme 20 preferably includes the user's personal preferences, and information regarding the user's viewing history such as for example browsing history, filtering history, searching history, and device setting history. The user's personal preferences includes information regarding particular programs and categorizations of programs that the user prefers to view. The user description scheme may also include personal information about the particular user, such as demographic and geographic information, e.g. zip code and age. The explicit definition of the particular programs or attributes related thereto permits the system 16 to select those programs from the information contained within the available program description schemes 18 that may be of interest to the user. Frequently, the user does not desire to learn to program the device nor desire to explicitly program the device. In addition, the user description scheme 20 may not be sufficiently robust to include explicit definitions describing all desirable programs for a particular user. In such a case, the capability of the user description scheme 20 to adapt to the viewing habits of the user to accommodate different viewing characteristics not explicitly provided for or otherwise difficult to describe is useful. In such a case, the user description scheme 20 may be augmented or any technique can be used to compare the information contained in the user description scheme 20 to the available information contained in the program description scheme 18 to make selections. The user description scheme provides a technique for holding user preferences ranging from program categories to program views, as well as usage history. User description scheme information is persistent but can be updated by the user or by an intelligent software agent on behalf of the user at any arbitrary time. It may also be

disabled by the user, at any time, if the user decides to do so. In addition, the user description scheme is modular and portable so that users can carry or port it from one device to another, such as with a handheld electronic device or smart card or transported over a network connecting multiple devices. When user description scheme is standardized among different manufacturers or products, user preferences become portable. For example, a user can personalize the television receiver in a hotel room permitting users to access information they prefer at any time and anywhere. In a sense, the user description scheme is persistent and timeless based. In addition, selected information within the program description scheme may be encrypted since at least part of the information may be deemed to be private (e.g., demographics). A user description scheme may be associated with an audiovisual program broadcast and compared with a particular user's description scheme of the receiver to readily determine whether or not the program's intended audience profile matches that of the user. It is to be understood that in one of the embodiments of the invention merely the user description scheme is included.

The system description scheme 22 preferably manages the individual programs and other data. The management may include maintaining lists of programs, categories, channels, users, videos, audio, and images. The management may include the capabilities of a device for providing the audio, video, and/or images. Such capabilities may include, for example, screen size, stereo, AC3, DTS, color, black/white, etc. The management may also include relationships between any one or more of the user, the audio, and the images in relation to one or more of a program description scheme(s) and a

user description scheme(s). In a similar manner the management may include relationships between one or more of the program description scheme(s) and user description scheme(s). It is to be understood that in one of the embodiments of the invention merely the system description scheme is included.

5 The descriptors of the program description scheme and the user description scheme should overlap, at least partially, so that potential desirability of the program can be determined by comparing descriptors representative of the same information. For example, the program and user description scheme may include the same set of categories and actors. The program description scheme has no knowledge of
10 the user description scheme, and vice versa, so that each description scheme is not dependant on the other for its existence. It is not necessary for the description schemes to be fully populated. It is also beneficial not to include the program description scheme with the user description scheme because there will likely be thousands of programs with associated description schemes which if combined with the user description scheme
15 would result in a unnecessarily large user description scheme. It is desirable to maintain the user description scheme small so that it is more readily portable. Accordingly, a system including only the program description scheme and the user description scheme would be beneficial.

20 The user description scheme and the system description scheme should include at least partially overlapping fields. With overlapping fields the system can capture the desired information, which would otherwise not be recognized as desirable. The system description scheme preferably includes a list of users and available programs.

Based on the master list of available programs, and associated program description scheme, the system can match the desired programs. It is also beneficial not to include the system description scheme with the user description scheme because there will likely be thousands of programs stored in the system description schemes which if combined with the user description scheme would result in a unnecessarily large user description scheme. It is desirable to maintain the user description scheme small so that it is more readily portable. For example, the user description scheme may include radio station preselected frequencies and/or types of stations, while the system description scheme includes the available stations for radio stations in particular cities. When traveling to a different city the user description scheme together with the system description scheme will permit reprogramming the radio stations. Accordingly, a system including only the system description scheme and the user description scheme would be beneficial.

The program description scheme and the system description scheme should include at least partially overlapping fields. With the overlapping fields, the system description scheme will be capable of storing the information contained within the program description scheme, so that the information is properly indexed. With proper indexing, the system is capable of matching such information with the user information, if available, for obtaining and recording suitable programs. If the program description scheme and the system description scheme were not overlapping then no information would be extracted from the programs and stored. System capabilities specified within the system description scheme of a particular viewing system can be correlated with a program description scheme to determine the views that can be supported by the viewing

system. For instance, if the viewing device is not capable of playing back video, its system description scheme may describe its viewing capabilities as limited to keyframe view and slide view only. Program description scheme of a particular program and system description scheme of the viewing system are utilized to present the appropriate views to the viewing system. Thus, a server of programs serves the appropriate views according to a particular viewing system's capabilities, which may be communicated over a network or communication channel connecting the server with user's viewing device. It is preferred to maintain the program description scheme separate from the system description scheme because the content providers repackage the content and description schemes in different styles, times, and formats. Preferably, the program description scheme is associated with the program, even if displayed at a different time. Accordingly, a system including only the system description scheme and the program description scheme would be beneficial.

By preferably maintaining the independence of each of the three description schemes while having fields that correlate the same information, the programs 10, the users 14, and the system 12 may be interchanged with one another while maintaining the functionality of the entire system 16. Referring to FIG. 2, the audio, visual, or audiovisual program 38, is received by the system 16. The program 38 may originate at any suitable source, such as for example broadcast television, cable television, satellite television, digital television, Internet broadcasts, world wide web, digital video discs, still images, video cameras, laser discs, magnetic media, computer hard drive, video tape, audio tape, data services, radio broadcasts, and microwave

communications. The program description stream may originate from any suitable source, such as for example PSIP/DVB-SI information in digital television broadcasts, specialized digital television data services, specialized Internet services, world wide web, data files, data over the telephone, and memory, such as computer memory. The program, user, and/or system description scheme may be transported over a network (communication channel). For example, the system description scheme may be transported to the source to provide the source with views or other capabilities that the device is capable of using. In response, the source provides the device with image, audio, and/or video content customized or otherwise suitable for the particular device. The system 16 may include any device(s) suitable to receive any one or more of such programs 38. An audiovisual program analysis module 42 performs an analysis of the received programs 38 to extract and provide program related information (descriptors) to the description scheme (DS) generation module 44. The program related information may be extracted from the data stream including the program 38 or obtained from any other source, such as for example data transferred over a telephone line, data already transferred to the system 16 in the past, or data from an associated file. The program related information preferably includes data defining both the program views and the program profiles available for the particular program 38. The analysis module 42 performs an analysis of the programs 38 using information obtained from (i) automatic audio-video analysis methods on the basis of low-level features that are extracted from the program(s), (ii) event detection techniques, (iii) data that is available (or extractable) from data sources or electronic program guides (EPGs, DVB-SI, and PSIP), and (iv) user

information obtained from the user description scheme 20 to provide data defining the program description scheme.

The selection of a particular program analysis technique depends on the amount of readily available data and the user preferences. For example, if a user prefers to watch a 5 minute video highlight of a particular program, such as a basketball game, the analysis module 42 may invoke a knowledge based system 90 (FIG. 3) to determine the highlights that form the best 5 minute summary. The knowledge based system 90 may invoke a commercial filter 92 to remove commercials and a slow motion detector 54 to assist in creating the video summary. The analysis module 42 may also invoke other modules to bring information together (e.g., textual information) to author particular program views. For example, if the program 38 is a home video where there is no further information available then the analysis module 42 may create a key-frame summary by identifying key-frames of a multi-level summary and passing the information to be used to generate the program views, and in particular a key frame view, to the description scheme. Referring also to FIG. 3, the analysis module 42 may also include other sub-modules, such as for example, a de-mux/decoder 60, a data and service content analyzer 62, a text processing and text summary generator 64, a close caption analyzer 66, a title frame generator 68, an analysis manager 70, an audiovisual analysis and feature extractor 72, an event detector 74, a key-frame summarizer 76, and a highlight summarizer 78.

The generation module 44 receives the system information 46 for the system description scheme. The system information 46 preferably includes data for the

system description scheme 22 generated by the generation module 44. The generation module 44 also receives user information 48 including data for the user description scheme. The user information 48 preferably includes data for the user description scheme generated within the generation module 44. The user input 48 may include, for example, meta information to be included in the program and system description scheme. The user description scheme (or corresponding information) is provided to the analysis module 42 for selective analysis of the program(s) 38. For example, the user description scheme may be suitable for triggering the highlight generation functionality for a particular program and thus generating the preferred views and storing associated data in the program description scheme. The generation module 44 and the analysis module 42 provide data to a data storage unit 50. The storage unit 50 may be any storage device, such as memory or magnetic media.

A search, filtering, and browsing (SFB) module 52 implements the description scheme technique by parsing and extracting information contained within the description scheme. The SFB module 52 may perform filtering, searching, and browsing of the programs 38, on the basis of the information contained in the description schemes. An intelligent software agent is preferably included within the SFB module 52 that gathers and provides user specific information to the generation module 44 to be used in authoring and updating the user description scheme (through the generation module 44). In this manner, desirable content may be provided to the user through a display 80. The selections of the desired program(s) to be retrieved, stored, and/or viewed may be programmed, at least in part, through a graphical user interface 82. The graphical user

interface may also include or be connected to a presentation engine for presenting the information to the user through the graphical user interface.

The intelligent management and consumption of audiovisual information using the multi-part description stream device provides a next-generation device suitable for the modern era of information overload. The device responds to changing lifestyles of individuals and families, and allows everyone to obtain the information they desire anytime and anywhere they want.

An example of the use of the device may be as follows. A user comes home from work late Friday evening being happy the work week is finally over. The user desires to catch up with the events of the world and then watch ABC's 20/20 show later that evening. It is now 9 PM and the 20/20 show will start in an hour at 10 PM. The user is interested in the sporting events of the week, and all the news about the Microsoft case with the Department of Justice. The user description scheme may include a profile indicating a desire that the particular user wants to obtain all available information regarding the Microsoft trial and selected sporting events for particular teams. In addition, the system description scheme and program description scheme provide information regarding the content of the available information that may selectively be obtained and recorded. The system, in an autonomous manner, periodically obtains and records the audiovisual information that may be of interest to the user during the past week based on the three description schemes. The device most likely has recorded more than one hour of audiovisual information so the information needs to be condensed in some manner. The user starts interacting with the system with a pointer or voice

commands to indicate a desire to view recorded sporting programs. On the display, the user is presented with a list of recorded sporting events including Basketball and Soccer. Apparently the user's favorite Football team did not play that week because it was not recorded. The user is interested in basketball games and indicates a desire to view games. A set of title frames is presented on the display that captures an important moment of each game. The user selects the Chicago Bulls game and indicates a desire to view a 5 minute highlight of the game. The system automatically generates highlights. The highlights may be generated by audio or video analysis, or the program description scheme includes data indicating the frames that are presented for a 5 minute highlight. The system may have also recorded web-based textual information regarding the particular Chicago-Bulls game which may be selected by the user for viewing. If desired, the summarized information may be recorded onto a storage device, such as a DVD with a label. The stored information may also include an index code so that it can be located at a later time. After viewing the sporting events the user may decide to read the news about the Microsoft trial. It is now 9:50 PM and the user is done viewing the news. In fact, the user has selected to delete all the recorded news items after viewing them. The user then remembers to do one last thing before 10 PM in the evening. The next day, the user desires to watch the VHS tape that he received from his brother that day, containing footage about his brother's new baby girl and his vacation to Peru last summer. The user wants to watch the whole 2-hour tape but he is anxious to see what the baby looks like and also the new stadium built in Lima, which was not there last time he visited Peru. The user plans to take a quick look at a visual summary of the tape, browse, and perhaps

watch a few segments for a couple of minutes, before the user takes his daughter to her piano lesson at 10 AM the next morning. The user plugs in the tape into his VCR, that is connected to the system, and invokes the summarization functionality of the system to scan the tape and prepare a summary. The user can then view the summary the next morning to quickly discover the baby's looks, and playback segments between the key-frames of the summary to catch a glimpse of the crying baby. The system may also record the tape content onto the system hard drive (or storage device) so the video summary can be viewed quickly. It is now 10:10 PM, and it seems that the user is 10 minutes late for viewing 20/20. Fortunately, the system, based on the three description schemes, has already been recording 20/20 since 10 PM. Now the user can start watching the recorded portion of 20/20 as the recording of 20/20 proceeds. The user will be done viewing 20/20 at 11:10 PM.

The average consumer has an ever increasing number of multimedia devices, such as a home audio system, a car stereo, several home television sets, web browsers, etc. The user currently has to customize each of the devices for optimal viewing and/or listening preferences. By storing the user preferences on a removable storage device, such as a smart card, the user may insert the card including the user preferences into such media devices for automatic customization. This results in the desired programs being automatically recorded on the VCR, and setting of the radio stations for the car stereo and home audio system. In this manner the user only has to specify his preferences at most once, on a single device and subsequently, the descriptors are automatically uploaded into devices by the removable storage device. The user

description scheme may also be loaded into other devices using a wired or wireless network connection, e.g. that of a home network. Alternatively, the system can store the user history and create entries in the user description scheme based on the's audio and video viewing habits. In this manner, the user would never need to program the viewing information to obtain desired information. In a sense, the user descriptor scheme enables modeling of the user by providing a central storage for the user's listening, viewing, browsing preferences, and user's behavior. This enables devices to be quickly personalized, and enables other components, such as intelligent agents, to communicate on the basis of a standardized description format, and to make smart inferences regarding the user's preferences.

Many different realizations and applications can be readily derived from FIGS. 2 and 3 by appropriately organizing and utilizing their different parts, or by adding peripherals and extensions as needed. In its most general form, FIG. 2 depicts an audiovisual searching, filtering, browsing, and/or recording appliance that is personalizable. The list of more specific applications/implementations given below is not exhaustive but covers a range.

The user description scheme is a major enabler for personalizable audiovisual appliances. If the structure (syntax and semantics) of the description schemes is known amongst multiple appliances, the user (user) can carry (or otherwise transfer) the information contained within his user description scheme from one appliance to another, perhaps via a smart card--where these appliances support smart card interface--in order to personalize them. Personalization can range from device settings, such as

display contrast and volume control, to settings of television channels, radio stations, web stations, web sites, geographic information, and demographic information such as age, zip code etc. Appliances that can be personalized may access content from different sources. They may be connected to the web, terrestrial or cable broadcast, etc., and they may also access multiple or different types of single media such as video, music, etc.

For example, one can personalize the car stereo using a smart card plugged out of the home system and plugged into the car stereo system to be able to tune to favorite stations at certain times. As another example, one can also personalize television viewing, for example, by plugging the smart card into a remote control that in turn will autonomously command the television receiving system to present the user information about current and future programs that fits the user's preferences. Different members of the household can instantly personalize the viewing experience by inserting their own smart card into the family remote. In the absence of such a remote, this same type of personalization can be achieved by plugging in the smart card directly to the television system. The remote may likewise control audio systems. In another implementation, the television receiving system holds user description schemes for multiple users (users) in local storage and identify different users (or group of users) by using an appropriate input interface. For example an interface using user-voice identification technology. It is noted that in a networked system the user description scheme may be transported over the network.

The user description scheme is generated by direct user input, and by using a software that watches the user to determine his/her usage pattern and usage

history. User description scheme can be updated in a dynamic fashion by the user or automatically. A well defined and structured description scheme design allows different devices to interoperate with each other. A modular design also provides portability.

The description scheme adds new functionality to those of the current VCR. An advanced VCR system can learn from the user via direct input of preferences, or by watching the usage pattern and history of the user. The user description scheme holds user's preferences users and usage history. An intelligent agent can then consult with the user description scheme and obtain information that it needs for acting on behalf of the user. Through the intelligent agent, the system acts on behalf of the user to discover programs that fit the taste of the user, alert the user about such programs, and/or record them autonomously. An agent can also manage the storage in the system according to the user description scheme, i.e., prioritizing the deletion of programs (or alerting the user for transfer to a removable media), or determining their compression factor (which directly impacts their visual quality) according to user's preferences and history.

The program description scheme and the system description scheme work in collaboration with the user description scheme in achieving some tasks. In addition, the program description scheme and system description scheme in an advanced VCR or other system will enable the user to browse, search, and filter audiovisual programs. Browsing in the system offers capabilities that are well beyond fast forwarding and rewinding. For instance, the user can view a thumbnail view of different categories of programs stored in the system. The user then may choose frame view, shot view, key frame view, or highlight view, depending on their availability and user's preference.

These views can be readily invoked using the relevant information in the program description scheme, especially in program views. The user at any time can start viewing the program either in parts, or in its entirety.

In this application, the program description scheme may be readily available from many services such as: (i) from broadcast (carried by EPG defined as a part of ATSC-PSIP (ATSC-Program Service Integration Protocol) in USA or DVB-SI (Digital Video Broadcast-Service Information) in Europe); (ii) from specialized data services (in addition to PSIP/DVB-SI); (iii) from specialized web sites; (iv) from the media storage unit containing the audiovisual content (e.g., DVD); (v) from advanced cameras (discussed later), and/or may be generated (i.e., for programs that are being stored) by the analysis module 42 or by user input 48.

Contents of digital still and video cameras can be stored and managed by a system that implements the description schemes, e.g., a system as shown in FIG. 2.

Advanced cameras can store a program description scheme, for instance, in addition to the audiovisual content itself. The program description scheme can be generated either in part or in its entirety on the camera itself via an appropriate user input interface (e.g., speech, visual menu drive, etc.). Users can input to the camera the program description scheme information, especially those high-level (or semantic) information that may otherwise be difficult to automatically extract by the system. Some camera settings and parameters (e.g., date and time), as well as quantities computed in the camera (e.g., color histogram to be included in the color profile), can also be used in generating the program description scheme. Once the camera is connected, the system can browse the camera

content, or transfer the camera content and its description scheme to the local storage for future use. It is also possible to update or add information to the description scheme generated in the camera.

5 The IEEE 1394 and Havi standard specifications enable this type of "audiovisual content" centric communication among devices. The description scheme API's can be used in the context of Havi to browse and/or search the contents of a camera or a DVD which also contain a description scheme associated with their content, i.e., doing more than merely invoking the PLAY API to play back and linearly view the media. The description schemes may be used in archiving audiovisual programs in a database. The search engine uses the information contained in the program description scheme to retrieve programs on the basis of their content. The program description scheme can also be used in navigating through the contents of the database or the query results. The user description scheme can be used in prioritizing the results of the user query during presentation. It is possible of course to make the program description scheme more comprehensive depending on the nature of the particular application.

 The description scheme fulfills the user's desire to have applications that pay attention and are responsive to their viewing and usage habits, preferences, and personal demographics. The proposed user description scheme directly addresses this desire in its selection of fields and interrelationship to other description schemes.

20 Because the description schemes are modular in nature, the user can port his user description scheme from one device to another in order to "personalize" the device.

The proposed description schemes can be incorporated into current products similar to those from TiVo and Replay TV in order to extend their entertainment informational value. In particular, the description scheme will enable audiovisual browsing and searching of programs and enable filtering within a particular program by supporting multiple program views such as the highlight view. In addition, the description scheme will handle programs coming from sources other than television broadcasts for which TiVo and Replay TV are not designed to handle. In addition, by standardization of TiVo and Replay TV type of devices, other products may be interconnected to such devices to extend their capabilities, such as devices supporting an MPEG 7 description. MPEG-7 is the Moving Pictures Experts Group - 7, acting to standardize descriptions and description schemes for audiovisual information. The device may also be extended to be personalized by multiple users, as desired.

Because the description scheme is defined, the intelligent software agents can communicate among themselves to make intelligent inferences regarding the user's preferences. In addition, the development and upgrade of intelligent software agents for browsing and filtering applications can be simplified based on the standardized user description scheme.

The description scheme is multi-modal in the following sense that it holds both high level (semantic) and low level features and/or descriptors. For example, the high and low level descriptors are actor name and motion model parameters, respectively. High level descriptors are easily readable by humans while low level descriptors are more easily read by machines and less understandable by humans. The program description

scheme can be readily harmonized with existing EPG, PSIP, and DVB-SI information facilitating search and filtering of broadcast programs. Existing services can be extended in the future by incorporating additional information using the compliant description scheme.

5 For example, one case may include audiovisual programs that are prerecorded on a media such as a digital video disc where the digital video disc also contains a description scheme that has the same syntax and semantics of the description scheme that the FSB module uses. If the FSB module uses a different description scheme, a transcoder (converter) of the description scheme may be employed. The user
10 may want to browse and view the content of the digital video disc. In this case, the user may not need to invoke the analysis module to author a program description. However, the user may want to invoke his or her user description scheme in filtering, searching and browsing the digital video disc content. Other sources of program information may likewise be used in the same manner.

15 It is to be understood that any of the techniques described herein with relation to video are equally applicable to images (such as still image or a frame of a video) and audio (such as radio).

An example of an audiovisual interface is shown in FIGS. 4-12 which is suitable for the preferred audiovisual description scheme. Referring to FIG. 4, by
20 selecting the thumbnail function as a function of category provides a display with a set of categories on the left hand side. Selecting a particular category, such as news, provides a set of thumbnail views of different programs that are currently available for viewing. In

addition, the different programs may also include programs that will be available at a different time for viewing. The thumbnail views are short video segments that provide an indication of the content of the respective actual program that it corresponds with.

Referring to FIG. 5, a thumbnail view of available programs in terms of channels may be displayed, if desired. Referring to FIG. 6, a text view of available programs in terms of channels may be displayed, if desired. Referring to FIG. 7, a frame view of particular programs may be displayed, if desired. A representative frame is displayed in the center of the display with a set of representative frames of different programs in the left hand column. The frequency of the number of frames may be selected, as desired. Also a set of frames are displayed on the lower portion of the display representative of different frames during the particular selected program. Referring to FIG. 8, a shot view of particular programs may be displayed, as desired. A representative frame of a shot is displayed in the center of the display with a set of representative frames of different programs in the left hand column. Also a set of shots are displayed on the lower portion of the display representative of different shots (segments of a program, typically sequential in nature) during the particular selected program. Referring to FIG. 9, a key frame view of particular programs may be displayed, as desired. A representative frame is displayed in the center of the display with a set of representative frames of different programs in the left hand column. Also a set of key frame views are displayed on the lower portion of the display representative of different key frame portions during the particular selected program. The number of key frames in each key frame view can be adjusted by selecting the level. Referring to FIG. 10, a highlight view may likewise be

displayed, as desired. Referring to FIG. 11, an event view may likewise be displayed, as desired. Referring to FIG. 12, a character/object view may likewise be displayed, as desired.

An example of the description schemes is shown below in XML. The description scheme may be implemented in any language and include any of the included descriptions (or more), as desired.

The proposed program description scheme includes three major sections for describing a video program. The first section identifies the described program. The second section defines a number of views which may be useful in browsing applications. The third section defines a number of profiles which may be useful in filtering and search applications. Therefore, the overall structure of the proposed description scheme is as follows:

```
<?XML version="1.0">
<!DOCTYPE MPEG-7 SYSTEM "mpeg-7.dtd">
<ProgramIdentity>
  <ProgramID> ... </ProgramID>
  <ProgramName> ... </ProgramName>
  <SourceLocation> ... </SourceLocation>
</ProgramIdentity>
<ProgramViews>
  <ThumbnailView> ... </ThumbnailView>
  <SlideView> ... </SlideView>
  <FrameView> ... </FrameView>
```

<ShotView> ... </ShotView>

<KeyFrameView> ... </KeyFrameView>

<HighlightView> ... </HighlightView>

<EventView> ... </EventView>

5 <CloseUpView> ... </CloseUpView>

<AlternateView> ... </AlternateView>

</ProgramViews>

<ProgramProfiles>

<GeneralProfile> ... </GeneralProfile>

10 <CategoryProfile> ... </CategoryProfile>

<DateTimeProfile> ... </DateTimeProfile>

<KeywordProfile> ... </KeywordProfile>

<TriggerProfile> ... </TriggerProfile>

<StillProfile> ... </StillProfile>

15 <EventProfile> ... </EventProfile>

<CharacterProfile> ... </CharacterProfile>

<ObjectProfile> ... </ObjectProfile>

<ColorProfile> ... </ColorProfile>

<TextureProfile> ... </TextureProfile>

20 <ShapeProfile> ... </ShapeProfile>

<MotionProfile> ... </MotionProfile>

</ProgramProfiles>

Program Identity

- Program ID

25

<ProgramID> program-id </ProgramID>

The descriptor <ProgramID> contains a number or a string to identify a program.

- **Program name**

<ProgramName> program-name </ProgramName>

5

The descriptor <ProgramName> specifies the name of a program.

- **Source location**

<SourceLocation> source-url </SourceLocation>

10

The descriptor <SourceLocation> specifies the location of a program in URL format.

Program Views

- **Thumbnail view**

15

<ThumbnailView>

<Image> thumbnail-image </Image>

</ThumbnailView>

The descriptor <ThumbnailView> specifies an image as the thumbnail representation of a program.

- **Slide view**

5 <SlideView> frame-id ... </SlideView>

The descriptor <SlideView> specifies a number of frames in a program which may be viewed as snapshots or in a slide show manner.

- **Frame view**

10

<FrameView> start-frame-id end-frame-id </FrameView>

The descriptor <FrameView> specifies the start and end frames of a program. This is the most basic view of a program and any program has a frame view.

15

- **Shot view**

<ShotView>

<Shot id=""> start-frame-id end-frame-id display-frame-id </Shot>

<Shot id=""> start-frame-id end-frame-id display-frame-id </Shot>

...

</ShotView>

5 The descriptor <ShotView> specifies a number of shots in a program. The <Shot> descriptor defines the start and end frames of a shot. It may also specify a frame to represent the shot.

- **Key-frame view**

<KeyFrameView>

10 <KeyFrames level="">

 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

 ...

 </KeyFrames>

15 <KeyFrames level="">

 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

 ...

 </KeyFrames>

20 ...

</KeyFrameView>

The descriptor <KeyFrameView> specifies key frames in a program. The key frames may be organized in a hierarchical manner and the hierarchy is captured by the descriptor

<KeyFrames> with a level attribute. The clips which are associated with each key frame are defined by the descriptor <Clip>. Here the display frame in each clip is the corresponding key frame.

- **Highlight view**

```

<HighlightView>
  <Highlight length="">
    <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>
    <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>
    ...
  </Highlight>
  <Highlight length="">
    <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>
    <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>
    ...
  </Highlight>
  ...
</HighlightView>

```

The descriptor <HighlightView> specifies clips to form highlights of a program. A program may have different versions of highlights which are tailored into various time length. The clips are grouped into each version of highlight which is specified by the descriptor <Highlight> with a length attribute.

- **Event view**

<EventView>

<Events name="">

5 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

<Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

...

</Events>

<Events name="">

10 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

<Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

...

</Events>

...

15 </EventView>

The descriptor <EventView> specifies clips which are related to certain events in a program. The clips are grouped into the corresponding events which are specified by the descriptor <Event> with a name attribute.

20 • **Close-up view**

<CloseUpView>

<Target name="">

<Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

<Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

...

</Target>

<Target name="">

5 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

 <Clip id=""> start-frame-id end-frame-id display-frame-id </Clip>

...

</Target>

...

10 </CloseUpView>

The descriptor <CloseUpView> specifies clips which may be zoomed in to certain targets in a program. The clips are grouped into the corresponding targets which are specified by the descriptor <Target> with a name attribute.

15 • **Alternate view**

<AlternateView>

 <AlternateSource id=""> source-url </AlternateSource>

 <AlternateSource id=""> source-url </AlternateSource>

20 ...

</AlternateView>

The descriptor <AlternateView> specifies sources which may be shown as alternate views of a program. Each alternate view is specified by the descriptor <AlternateSource> with an id attribute. The locate of the source may be specified in URL format.

Program Profiles

• General profile

```

<GeneralProfile>
    <Title> title-text </Title>
    <Abstract> abstract-text </Abstract>
    <Audio> voice-annotation </Audio>
    <Www> web-page-url </Www>
    <ClosedCaption> yes/no </ClosedCaption>
    <Language> language-name </Language>
    <Rating> rating </Rating>
    <Length> time </Length>
    <Authors> author-name ... </Authors>
    <Producers> producer-name ... </Producers>
    <Directors> director-name ... </Directors>
    <Actors> actor-name ... </Actors>
    ...
</GeneralProfile>

```

The descriptor <GeneralProfile> describes the general aspects of a program.

- **Category profile**

<CategoryProfile> category-name ... </CategoryProfile>

5 The descriptor <CategoryProfile> specifies the categories under which a program may be classified.

- **Date-time profile**

<DateTimeProfile>

10 <ProductionDate> date </ProductionDate>

 <ReleaseDate> date </ReleaseDate>

 <RecordingDate> date </RecordingDate>

 <RecordingTime> time </RecordingTime>

 ...

15 </DateTimeProfile>

 The descriptor <DateTimeProfile> specifies various date and time information of a program.

- **Keyword profile**

20 <KeywordProfile> keyword ... </KeywordProfile>

The descriptor <KeywordProfile> specifies a number of keywords which may be used to filter or search a program.

- **Trigger profile**

5

```
<TriggerProfile> trigger-frame-id ... </TriggerProfile>
```

The descriptor <TriggerProfile> specifies a number of frames in a program which may be used to trigger certain actions while the playback of the program.

10

- **Still profile**

```
<StillProfile>
```

```
  <Still id="">
```

```
    <HotRegion id="">
```

15

```
      <Location> x1 y1 x2 y2 </Location>
```

```
      <Text> text-annotation </Text>
```

```
      <Audio> voice-annotation </Audio>
```

```
      <Www> web-page-url </Www>
```

```
    </HotRegion>
```

20

```
  <HotRegion id="">
```

```
    <Location> x1 y1 x2 y2 </Location>
```

```
    <Text> text-annotation </Text>
```

```

        <Audio> voice-annotation </Audio>

        <Www> web-page-url </Www>

    </HotRegion>

    ...

5    </Still>

    <Still id="">

        <HotRegion id="">

            <Location> x1 y1 x2 y2 </Location>

            <Text> text-annotation </Text>

10   <Audio> voice-annotation </Audio>

        <Www> web-page-url </Www>

    </HotRegion>

    <HotRegion id="">

        <Location> x1 y1 x2 y2 </Location>

15   <Text> text-annotation </Text>

        <Audio> voice-annotation </Audio>

        <Www> web-page-url </Www>

    </HotRegion>

    ...

20   </Still>

    ...

</StillProfile>

```

The descriptor <StillProfile> specifies hot regions or regions of interest within a frame.

The frame is specified by the descriptor <Still> with an id attribute which corresponds to the frame-id. Within a frame, each hot region is specified by the descriptor <HotRegion> with an id attribute.

- **Event profile**

<EventProfile>

<EventList> event-name ... </EventList>

5

<Event name="">

<Www> web-page-url </Www>

<Occurrence id="">

<Duration> start-frame-id end-frame-id </Duration>

<Text> text-annotation </Text>

10

<Audio> voice-annotation </Audio>

</Occurrence>

<Occurrence id="">

<Duration> start-frame-id end-frame-id </Duration>

<Text> text-annotation </Text>

15

<Audio> voice-annotation </Audio>

</Occurrence>

...

</Event>

<Event name="">

20

<Www> web-page-url </Www>

<Occurrence id="">

<Duration> start-frame-id end-frame-id </Duration>

<Text> text-annotation </Text>

<Audio> voice-annotation </Audio>

25

</Occurrence>

<Occurrence id="">

<Duration> start-frame-id end-frame-id </Duration>

<Text> text-annotation </Text>

```

        <Audio> voice-annotation </Audio>

    </Occurrence>

    ...

</Event>

5    ...

</EventProfile>

```

The descriptor <EventProfile> specifies the detailed information for certain events in a program. Each event is specified by the descriptor <Event> with a name attribute. Each occurrence of an event is specified by the descriptor <Occurrence> with an id attribute which may be matched with a clip id under <EventView>.

- **Character profile**

```

<CharacterProfile>

15    <CharacterList> character-name ... </CharacterList>

    <Character name="">

        <ActorName> actor-name </ActorName>

        <Gender> male </Gender>

        <Age> age </Age>

20    <Www> web-page-url </Www>

    <Occurrence id="">

        <Duration> start-frame-id end-frame-id </Duration>

        <Location> frame:[x1 y1 x2 y2] ... </Location>

        <Motion> vx vy vz va vp vγ </Motion>

25    <Text> text-annotation </Text>

```

```

    <Audio> voice-annotation </Audio>

</Occurrence>

<Occurrence id="">

    <Duration> start-frame-id end-frame-id </Duration>

    <Location> frame:[x1 y1 x2 y2] ... </Location>

    <Motion> vx vy vz vα vβ vγ </Motion>

    <Text> text-annotation </Text>

    <Audio> voice-annotation </Audio>

</Occurrence>

...

</Character>

<Character name="">

    <ActorName> actor-name </ActorName>

    <Gender> male </Gender>

    <Age> age </Age>

    <Www> web-page-url </Www>

    <Occurrence id="">

        <Duration> start-frame-id end-frame-id </Duration>

        <Location> frame:[x1 y1 x2 y2] ... </Location>

        <Motion> vx vy vz vα vβ vγ </Motion>

        <Text> text-annotation </Text>

        <Audio> voice-annotation </Audio>

    </Occurrence>

    <Occurrence id="">

        <Duration> start-frame-id end-frame-id </Duration>

        <Location> frame:[x1 y1 x2 y2] ... </Location>

        <Motion> vx vy vz vα vβ vγ </Motion>

        <Text> text-annotation </Text>

        <Audio> voice-annotation </Audio>

```

</Occurrence>

...

</Character>

...

5 </CharacterProfile>

The descriptor <CharacterProfile> specifies the detailed information for certain characters in a program. Each character is specified by the descriptor <Character> with a name attribute. Each occurrence of a character is specified by the descriptor <Occurrence> with an id attribute which may be matched with a clip id under <CloseUpView>.

- **Object profile**

<ObjectProfile>

<ObjectList> object-name ... </ObjectList>

<Object name="">

<Www> web-page-url </Www>

<Occurrence id="">

<Duration> start-frame-id end-frame-id </Duration>

<Location> frame:[x1 y1 x2 y2] ... </Location>

<Motion> v_x v_y v_z v_α v_β v_γ </Motion>

<Text> text-annotation </Text>

<Audio> voice-annotation </Audio>

</Occurrence>

```

<Occurrence id="">
    <Duration> start-frame-id end-frame-id </Duration>
    <Location> frame:[x1 y1 x2 y2] ... </Location>
    <Motion> vx vy vz va vβ vγ </Motion>
    <Text> text-annotation </Text>
    <Audio> voice-annotation </Audio>
</Occurrence>

```

...

```

</Object>

```

```

<Object name="">
    <Www> web-page-url </Www>

```

```

<Occurrence id="">
    <Duration> start-frame-id end-frame-id </Duration>
    <Location> frame:[x1 y1 x2 y2] ... </Location>
    <Motion> vx vy vz va vβ vγ </Motion>
    <Text> text-annotation </Text>
    <Audio> voice-annotation </Audio>
</Occurrence>

```

```

<Occurrence id="">
    <Duration> start-frame-id end-frame-id </Duration>
    <Location> frame:[x1 y1 x2 y2] ... </Location>
    <Motion> vx vy vz va vβ vγ </Motion>
    <Text> text-annotation </Text>
    <Audio> voice-annotation </Audio>

```

```

</Occurrence>

```

...

```

</Object>

```

...

```

</ObjectProfile>

```

The descriptor <ObjectProfile> specifies the detailed information for certain objects in a program. Each object is specified by the descriptor <Object> with a name attribute. Each occurrence of a object is specified by the descriptor <Occurrence> with an id attribute which may be matched with a clip id under <CloseUpView>.

- **Color profile**

<ColorProfile>

...

</ColorProfile>

The descriptor <ColorProfile> specifies the detailed color information of a program. All MPEG-7 color descriptors may be placed under here.

- **Texture profile**

<TextureProfile>

...

</TextureProfile>

The descriptor <TextureProfile> specifies the detailed texture information of a program. All MPEG-7 texture descriptors may be placed under here.

- **Shape profile**

<ShapeProfile>

...

5 </ShapeProfile>

The descriptor <ShapeProfile> specifies the detailed shape information of a program. All MPEG-7 shape descriptors may be placed under here.

- **Motion profile**

<MotionProfile>

...

</MotionProfile>

The descriptor <MotionProfile> specifies the detailed motion information of a program. All MPEG-7 motion descriptors may be placed under here.

User Description Scheme

The proposed user description scheme includes three major sections for describing a user. The first section identifies the described user. The second section records a number of settings which may be preferred by the user. The third section records some statistics which

may reflect certain usage patterns of the user. Therefore, the overall structure of the proposed description scheme is as follows:

```
<?XML version="1.0">
<!DOCTYPE MPEG-7 SYSTEM "mpeg-7.dtd">
5  <UserIdentity>
    <UserID> ... </UserID>
    <UserName> ... </UserName>
  </UserIdentity>
  <UserPreferences>
10  <BrowsingPreferences> ... </BrowsingPreferences>
    <FilteringPreferences> ... </FilteringPreferences>
    <SearchPreferences> ... </SearchPreferences>
    <DevicePreferences> ... </DevicePreferences>
  </UserPreferences>
15  <UserHistory>
    <BrowsingHistory> ... </BrowsingHistory>
    <FilteringHistory> ... </FilteringHistory>
    <SearchHistory> ... </SearchHistory>
    <DeviceHistory> ... </DeviceHistory>
20  </UserHistory>
    <UserDemographics>
    <Age> ... </Age>
    <Gender> ... </Gender>
    <ZIP> ... </ZIP>
25  </UserDemographics>
```

User Identity

- **User ID**

<UserID> user-id </UserID>

5

The descriptor <UserID> contains a number or a string to identify a user.

- **User name**

<UserName> user-name </UserName>

10

The descriptor <UserName> specifies the name of a user.

User Preferences

- **Browsing preferences**

15

<BrowsingPreferences>

<Views>

<ViewCategory id=""> view-id ... </ViewCategory>

<ViewCategory id=""> view-id ... </ViewCategory>

...

</Views>

<FrameFrequency> frequency ...<FrameFrequency>

<ShotFrequency> frequency ...<ShotFrequency>

<KeyFrameLevel> level-id ...<KeyFrameLevel>

<HighlightLength> length ...<HighlightLength>

...

</BrowsingPreferences>

The descriptor <BrowsingPreferences> specifies the browsing preferences of a user. The user's preferred views are specified by the descriptor <Views>. For each category, the preferred views are specified by the descriptor <ViewCategory> with an id attribute which corresponds to the category id. The descriptor <FrameFrequency> specifies at what interval the frames should be displayed on a browsing slider under the frame view. The descriptor <ShotFrequency> specifies at what interval the shots should be displayed on a browsing slider under the shot view. The descriptor <KeyFrameLevel> specifies at what level the key frames should be displayed on a browsing slider under the key frame view. The descriptor <HighlightLength> specifies which version of the highlight should be shown under the highlight view.

- **Filtering preferences**

<FilteringPreferences>

<Categories> category-name ... </Categories>

<Channels> channel-number ... </Channels>

<Ratings> rating-id ... </Ratings>
 <Shows> show-name ... </Shows>
 <Authors> author-name ... </Authors>
 <Producers> producer-name ... </Producers>
 <Directors> director-name ... </Directors>
 <Actors> actor-name ... </Actors>
 <Keywords> keyword ... </Keywords>
 <Titles> title-text ... </Titles>
 ...

5

10

</FilteringPreferences>

The descriptor <FilteringPreferences> specifies the filtering related preferences of a user.

- **Search preferences**

<SearchPreferences>
 <Categories> category-name ... </Categories>
 <Channels> channel-number ... </Channels>
 <Ratings> rating-id ... </Ratings>
 <Shows> show-name ... </Shows>
 <Authors> author-name ... </Authors>
 <Producers> producer-name ... </Producers>
 <Directors> director-name ... </Directors>
 <Actors> actor-name ... </Actors>
 <Keywords> keyword ... </Keywords>

15

20

25

<Titles> title-text ... </Titles>

...

</SearchPreferences>

5 The descriptor <SearchPreferences> specifies the search related preferences of a user.

- **Device preferences**

<DevicePreferences>

<Brightness> brightness-value </Brightness>

<Contrast> contrast-value </Contrast>

<Volume> volume-value </Volume>

</DevicePreferences>

The descriptor <DevicePreferences> specifies the device preferences of a user.

Usage History

- **Browsing history**

<BrowsingHistory>

<Views>

<ViewCategory id=""> view-id ... </ViewCategory>

<ViewCategory id=""> view-id ... </ViewCategory>

```

...
</Views>

<FrameFrequency> frequency ...<FrameFrequency>

<ShotFrequency> frequency ...<ShotFrequency>

5  <KeyFrameLevel> level-id ...<KeyFrameLevel>

<HighlightLength> length ...<HighlightLength>

...

</BrowsingHistory>

```

10 The descriptor <BrowsingHistory> captures the history of a user's browsing related activities.

- **Filtering history**

```

<FilteringHistory>

15  <Categories> category-name ... </Categories>

    <Channels> channel-number ... </Channels>

    <Ratings> rating-id ... </Ratings>

    <Shows> show-name ... </Shows>

    <Authors> author-name ... </Authors>

20  <Producers> producer-name ... </Producers>

    <Directors> director-name ... </Directors>

    <Actors> actor-name ... </Actors>

    <Keywords> keyword ... </Keywords>

    <Titles> title-text ... </Titles>

25  ...

```

</FilteringHistory>

The descriptor <FilteringHistory> captures the history of a user's filtering related activities.

5 • Search history

<SearchHistory>

<Categories> category-name ... </Categories>

<Channels> channel-number ... </Channels>

<Ratings> rating-id ... </Ratings>

<Shows> show-name ... </Shows>

<Authors> author-name ... </Authors>

<Producers> producer-name ... </Producers>

<Directors> director-name ... </Directors>

<Actors> actor-name ... </Actors>

<Keywords> keyword ... </Keywords>

<Titles> title-text ... </Titles>

...

</SearchHistory>

The descriptor <SearchHistory> captures the history of a user's search related activities.

- **Device history**

<DeviceHistory>

<Brightness> brightness-value ... </Brightness>

<Contrast> contrast-value ... </Contrast>

<Volume> volume-value ... </Volume>

</DeviceHistory>

The descriptor <DeviceHistory> captures the history of a user's device related activities.

User demographics

- **Age**

<Age> age </Age>

5

The descriptor <Age> specifies the age of a user.

- **Gender**

<Gender> ... </Gender>

10

The descriptor <Gender> specifies the gender of a user.

- **ZIP code**

<ZIP> ... </ZIP>

15

The descriptor <ZIP> specifies the ZIP code of where a user lives.

System Description Scheme

The proposed system description scheme includes four major sections for describing a user.

The first section identifies the described system. The second section keeps a list of all known users. The third section keeps lists of available programs. The fourth section describes the

capabilities of the system. Therefore, the overall structure of the proposed description scheme is as follows:

```
<?XML version="1.0">
<!DOCTYPE MPEG-7 SYSTEM "mpeg-7.dtd">
<SystemIdentity>
  <SystemID> ... </SystemID>
  <SystemName> ... </SystemName>
  <SystemSerialNumber> ... </SystemSerialNumber>
</SystemIdentity>
<SystemUsers>
  <Users> ... </Users>
</SystemUsers>
<SystemPrograms>
  <Categories> ... </Categories>
  <Channels> ... </Channels>
  <Programs> ... </Programs>
</SystemPrograms>
<SystemCapabilities>
  <Views> ... </Views>
</SystemCapabilities>
```

System Identity

- **System ID**

<SystemID> system-id </SystemID>

5

The descriptor <SystemID> contains a number or a string to identify a video system or device.

- **System name**

<SystemName> system-name </SystemName>

10

The descriptor <SystemName> specifies the name of a video system or device.

- **System serial number**

<SystemSerialNumber> system-serial-number </SystemSerialNumber>

15

The descriptor <SystemSerialNumber> specifies the serial number of a video system or device.

System Users

- **Users**

<Users>

5

<User>

<UserID> user-id </UserID>

<UserName> user-name </UserName>

</User>

<User>

10

<UserID> user-id </UserID>

<UserName> user-name </UserName>

</User>

...

</Users>

15

The descriptor <SystemUsers> lists a number of users who have registered on a video system or device. Each user is specified by the descriptor <User>. The descriptor <UserID> specifies a number or a string which should match with the number or string specified in <UserID> in one of the user description schemes.

20

Programs in the System

- **Categories**

<Categories>

<Category>

<CategoryID> category-id </CategoryID>

<CategoryName> category-name </CategoryName>

<SubCategories> sub-category-id ... </SubCategories>

</Category>

<Category>

<CategoryID> category-id </CategoryID>

<CategoryName> category-name </CategoryName>

<SubCategories> sub-category-id ... </SubCategories>

</Category>

...

</Categories>

The descriptor <Categories> lists a number of categories which have been registered on a video system or device. Each category is specified by the descriptor <Category>. The major-sub relationship between categories is captured by the descriptor <SubCategories>.

• Channels

<Channels>

<Channel>

<ChannelID> channel-id </ChannelID>

<ChannelName> channel-name </ChannelName>

```

    <SubChannels> sub-channel-id ... </SubChannels>

  </Channel>

  <Channel>

    <ChannelID> channel-id </ChannelID>

    <ChannelName> channel-name </ChannelName>

    <SubChannels> sub-channel-id ... </SubChannels>

  </Channel>

  ...

</Channels>

```

The descriptor <Channels> lists a number of channels which have been registered on a video system or device. Each channel is specified by the descriptor <Channel>. The major-sub relationship between channels is captured by the descriptor <SubChannels>.

- Programs

```

<Programs>

  <CategoryPrograms>

    <CategoryID> category-id </CategoryID>

    <Programs> program-id ... </Programs>

  </CategoryPrograms>

  <CategoryPrograms>

    <CategoryID> category-id </CategoryID>

    <Programs> program-id ... </Programs>

  </CategoryPrograms>

  ...

```

```

<ChannelPrograms>
  <ChannelID> channel-id </ChannelID>
  <Programs> program-id ... </Programs>
</ChannelPrograms>

```

```

5  <ChannelPrograms>
    <ChannelID> channel-id </ChannelID>
    <Programs> program-id ... </Programs>
  </ChannelPrograms>

```

...

```

10 </Programs>

```

The descriptor <Programs> lists programs who are available on a video system or device. The programs are grouped under corresponding categories or channels. Each group of programs are specified by the descriptor <CategoryPrograms> or <ChannelPrograms>. Each program id contained in the descriptor <Programs> should match with the number or string specified in <ProgramID> in one of the program description schemes.

System Capabilities

- Views

```

20 <Views>
    <View>
      <ViewID> view-id </ViewID>
      <ViewName> view-name </ViewName>
    </View>
25 <View>
    <ViewID> view-id </ViewID>
    <ViewName> view-name </ViewName>
  </View>
  ...

```

</Views>

The descriptor <Views> lists views which are supported by a video system or device.

Each view is specified by the descriptor <View>. The descriptor <ViewName>

contains a string which should match with one of the following views used in the program description schemes: ThumbnailView, SlideView, FrameView, ShotView, KeyFrameView, HighlightView, EventView, and CloseUpView.

The present inventors came to the realization that the program description scheme may be further modified to provide additional capabilities. Referring to FIG. 13, the modified program description scheme 400 includes four separate types of information, namely, a syntactic structure description scheme 402, a semantic structure description scheme 404, a visualization description scheme 406, and a meta information description scheme 408. It is to be understood that in any particular system one or more of the description schemes may be included, as desired.

Referring to FIG. 14, the visualization description scheme 406 enables fast and effective browsing of video program (and audio programs) by allowing access to the necessary data, preferably in a one-step process. The visualization description scheme 406 provides for several different presentations of the video content (or audio), such as for example, a thumbnail view description scheme 410, a key frame view description scheme 412, a highlight view description scheme 414, an event view description scheme 416, a close-up view description scheme 418, and an alternative view description scheme 420. Other presentation techniques and description schemes may be added, as desired.

The thumbnail view description scheme 410 preferably includes an image 422 or reference to an image representative of the video content and a time reference 424 to the video. The key frame view description scheme 412 preferably includes a level indicator 426 and a time reference 428. The level indicator 426 accommodates the presentation of a different number of key frames for the same video portion depending on the user's preference. The highlight view description scheme 414 includes a length indicator 430 and a time reference 432. The length indicator 430 accommodates the presentation of a different highlight duration of a video depending on the user's preference. The event view description scheme 416 preferably includes an event indicator 434 for the selection of the desired event and a time reference 436. The close-up view description scheme 418 preferably includes a target indicator 438 and a time reference 440. The alternate view description scheme preferably includes a source indicator 442. To increase performance of the system it is preferred to specify the data which is needed to render such views in a centralized and straightforward manner. By doing so, it is then feasible to access the data in a simple one-step process without complex parsing of the video.

Referring to FIG. 15, the meta information description scheme 408 generally includes various descriptors which carry general information about a video (or audio) program such as the title, category, keywords, etc. Additional descriptors, such as those previously described, may be included, as desired.

Referring again to FIG. 13, the syntactic structure description scheme 402 specifies the physical structure of a video program (or audio), e.g., a table of contents. The physical features, may include for example, color, texture, motion, etc. The syntactic

structure description scheme 402 preferably includes three modules, namely a segment description scheme 450, a region description scheme 452, and a segment/region relation graph description scheme 454. The segment description scheme 450 may be used to define relationships between different portions of the video consisting of multiple frames of the video. A segment description scheme 450 may contain another segment description scheme 450 and/or shot description scheme to form a segment tree. Such a segment tree may be used to define a temporal structure of a video program. Multiple segment trees may be created and thereby create multiple table of contents. For example, a video program may be segmented into story units, scenes, and shots, from which the segment description scheme 450 may contain such information as a table of contents. The shot description scheme may contain a number of key frame description schemes, a mosaic description scheme(s), a camera motion description scheme(s), etc. The key frame description scheme may contain a still image description scheme which may in turn contains color and texture descriptors. It is noted that various low level descriptors may be included in the still image description scheme under the segment description scheme. Also, the visual descriptors may be included in the region description scheme which is not necessarily under a still image description scheme. On example of a segment description scheme 450 is shown in FIG. 16.

Referring to FIG. 17, the region description scheme 452 defines the interrelationships between groups of pixels of the same and/or different frames of the video. The region description scheme 452 may also contain geometrical features, color, texture features, motion features, etc.

Referring to FIG. 18, the segment/region relation graph description scheme 454 defines the interrelationships between a plurality of regions (or region description schemes), a plurality of segments (or segment description schemes), and/or a plurality of regions (or description schemes) and segments (or description schemes).

5 Referring again to FIG. 13, the semantic structure description scheme 404 is used to specify semantic features of a video program (or audio), e.g. semantic events. In a similar manner to the syntactic structure description scheme, the semantic structure description scheme 404 preferably includes three modules, namely an event description scheme 480, an object description scheme 482, and an event/object relation graph description scheme 484. The event description scheme 480 may be used to form
10 relationships between different events of the video normally consisting of multiple frames of the video. An event description scheme 480 may contain another event description scheme 480 to form a segment tree. Such an event segment tree may be used to define a semantic index table for a video program. Multiple event trees may be created and
15 thereby creating multiple index tables. For example, a video program may include multiple events, such as a basketball dunk, a fast break, and a free throw, and the event description scheme may contain such information as an index table. The event description scheme may also contain references which link the event to the corresponding segments and/or regions specified in the syntactic structure description scheme. On
20 example of an event description scheme is shown in FIG. 19.

Referring to FIG. 20, the object description scheme 482 defines the interrelationships between groups of pixels of the same and/or different frames of the

video representative of objects. The object description scheme 482 may contain another object description scheme and thereby form an object tree. Such an object tree may be used to define an object index table for a video program. The object description scheme may also contain references which link the object to the corresponding segments and/or regions specified in the syntactic structure description scheme.

Referring to FIG. 21, the event/object relation graph description scheme 484 defines the interrelationships between a plurality of events (or event description schemes), a plurality of objects (or object description schemes), and/or a plurality of events (or description schemes) and objects (or description schemes).

After further consideration, the present inventors came the realization that the particular design of the user preference description scheme is important to implement portability, while permitting adaptive updating, of the user preference description scheme. Moreover, the user preference description scheme should be readily usable by the system while likewise being suitable for modification based on the user's historical usage patterns. It is possible to collectively track all users of a particular device to build a database for the historical viewing preferences of the users of the device, and thereafter process the data dynamically to determine which content the users would likely desire. However, this implementation would require the storage of a large amount of data and the associated dynamic processing requirements to determine the user preferences. It is to be understood that the user preference description scheme may be used alone or in combination with other description scheme.

Referring to FIG. 22, to achieve portability and potentially decreased processing requirements the user preference description scheme 20 should be divided into at least two separate description schemes, namely, a usage preference description scheme 500 and a usage history description scheme 502. The usage preference description scheme 500, described in detail later, includes a description scheme of the user's audio and/or video consumption preferences. The usage preference description scheme 500 describes one or more of the following, depending on the particular implementation, (a) browsing preferences, (b) filtering preferences, (c) searching preferences, and (d) device preferences of the user. The type of preferences shown in the usage preference description scheme 500 are generally immediately usable by the system for selecting and otherwise using the available audio and/or video content. In other words, the usage preference description scheme 500 includes data describing audio and/or video consumption of the user. The usage history description scheme 502, described in detail later, includes a description scheme of the user's historical audio and/or video activity, such as browsing, device settings, viewing, and selection. The usage history description scheme 502 describes one or more of the following, depending on the particular implementation, (a) browsing history, (b) filtering history, (c) searching history, and (d) device usage history. The type of preferences shown in the usage history description scheme 502 are not generally immediately usable by the system for selecting and otherwise using the available audio and/or video content. The data contained in the usage history description scheme 502 may be considered generally "unprocessed", at least in comparison to the data contained in the usage preferences description scheme 500

because it generally contains the historical usage data of the audio and/or video content of the viewer.

In general, capturing the user's usage history facilitates "automatic" composition of user preferences by a machine, as desired. When updating the user preference description scheme 500 it is desirable that the usage history description scheme 502 be relatively symmetric to the usage preference description scheme 500. The symmetry permits more effective updating because less interpretation between the two description schemes is necessary in order to determine what data should be included in the preferences. Numerous algorithms can then be applied in utilization of the history information in deriving user preferences. For instance, statistics can be computed from the history and utilized for this purpose.

After consideration of the usage preference description 500 and the usage history description 502, the present inventors came to the realization that in the home environment many different users with different viewing and usage preferences may use the same device. For example, with a male adult preferring sports, a female adult preferring afternoon talk shows, and a three year old child preferring children's programming, the total information contained in the usage preference description 500 and the usage history description 502 will not be individually suitable for any particular user. The resulting composite data and its usage by the device is frustrating to the users because the device will not properly select and present audio and/or video content that is tailored to any particular user. To alleviate this limitation, the user preference description 20 may also include a user identification (user identifier) description 504. The user

identification description 504 includes an identification of the particular user that is using the device. By incorporating a user identification description 504 more than one user may use the device while maintaining a different or a unique set of data within the usage preference description 500 and the usage history description 502. Accordingly, the user identification description 504 associates the appropriate usage preference description(s) 500 and usage history description(s) 502 for the particular user identified by the user identification description 504. With multiple user identification descriptions 504, multiple entries within a single user identification description 504 identifying different users, and/or including the user identification description within the usage preference description 500 and/or usage history description 502 to provide the association therebetween, multiple users can readily use the same device while maintaining their individuality. Also, without the user identification description in the preferences and/or history, the user may more readily customize content anonymously. In addition, the user's user identification description 504 may be used to identify multiple different sets of usage preference descriptions 500 -- usage history descriptions 502, from which the user may select for present interaction with the device depending on usage conditions. The use of multiple user identification descriptions for the same user is useful when the user uses multiple different types of devices, such as a television, a home stereo, a business television, a hotel television, and a vehicle audio player, and maintains multiple different sets of preference descriptions. Further, the identification may likewise be used to identify groups of individuals, such as for example, a family. In addition, devices that are used on a temporary basis, such as those in hotel rooms or rental cars, the user

identification requirements may be overridden by employing a temporary session user identification assigned by such devices. In applications where privacy concerns may be resolved or are otherwise not a concern, the user identification description 504 may also contain demographic information of the user. In this manner, as the usage history description 502 increases during use over time, this demographic data and/or data regarding usage patterns may be made available to other sources. The data may be used for any purpose, such as for example, providing targeted advertising or programming on the device based on such data.

Referring to FIG. 23, periodically an agent 510 processes the usage history description(s) 502 for a particular user to "automatically" determine the particular user's preferences. In this manner, the user's usage preference description 500 is updated to reflect data stored in the usage history description 502. This processing by the agent 510 is preferably performed on a periodic basis so that during normal operation the usage history description 502 does not need to be processed, or otherwise queried, to determine the user's current browsing, filtering, searching, and device preferences. The usage preference description 500 is relatively compact and suitable for storage on a portable storage device, such as a smart card, for use by other devices as previously described.

Frequently, the user may be traveling away from home with his smart card containing his usage preference description 500. During such traveling the user will likely be browsing, filtering, searching, and setting device preferences of audio and/or video content on devices into which he provided his usage preference description 500. However, in some circumstances the audio and/or video content browsed, filtered,

searched, and device preferences of the user may not be typically what he is normally interested in. In addition, for a single device the user may desire more than one profile depending on the season, such as football season, basketball season, baseball season, fall, winter, summer, and spring. Accordingly, it may not be appropriate for the device to
5 create a usage history description 502 and thereafter have the agent 510 "automatically" update the user's usage preference description 500. This will in effect corrupt the user's usage preference description 500. Accordingly, the device should include an option that disables the agent 510 from updating the usage preference description 500. Alternatively, the usage preference description 500 may include one or more fields or data structures
10 that indicate whether or not the user desires the usage preference description 500 (or portions thereof) to be updated.

Referring to FIG. 24, the device may use the program descriptions provided by any suitable source describing the current and/or future audio and/or video content available from which a filtering agent 520 selects the appropriate content for the
15 particular user(s). The content is selected based upon the usage preference description for a particular user identification(s) to determine a list of preferred audio and/or video programs.

As it may be observed, with a relatively compact user preference description 500 the user's preferences are readily movable to different devices, such as a
20 personal video recorder, a TiVO player, a RePlay Networks player, a car audio player, or other audio and/or video appliance. Yet, the user preference description 500 may be

updated in accordance with the user's browsing, filtering, searching, and device preferences.

Referring to FIG. 25, the usage preference description 500 preferably includes three different categories of descriptions, depending on the particular implementation. The preferred descriptions include (a) browsing preferences description 530, (b) filtering and search preferences description, 532 and (c) device preferences description 534. The browsing preferences description 530 relates to the viewing preferences of audio and/or video programs. The filtering and search preferences description 532 relates to audio and/or video program level preferences. The program level preferences are not necessarily used at the same time as the (browsing) viewing preferences. For example, preferred programs can be determined as a result of filtering program descriptions according to user's filtering preferences. A particular preferred program may subsequently be viewed in accordance with user's browsing preferences. Accordingly, efficient implementation may be achieved if the browsing preferences description 530 is separate, at least logically, from the filtering and search preferences description 532. The device preferences description 534 relates to the preferences for setting up the device in relation to the type of content being presented, e.g. romance, drama, action, violence, evening, morning, day, weekend, weekday, and/or the available presentation devices. For example, presentation devices may include stereo sound, mono sound, surround sound, multiple potential displays, multiple different sets of audio speakers, AC-3, and Dolby Digital. It may likewise be observed that the device

preferences description 534 is likewise separate, at least logically, from the browsing description 530 and filtering/search preferences description 532.

The browsing preferences description 530 contains descriptors that describe preferences of the user for browsing multimedia (audio and/or video) information. In the case of video, for example, the browsing preferences may include user's preference for continuous playback of the entire program versus visualizing a short summary of the program. Various summary types may be described in the program descriptions describing multiple different views of programs where these descriptions are utilized by the device to facilitate rapid non-linear browsing, viewing, and navigation. Parameters of the various summary types should also be specified, i.e., number of hierarchy levels when the keyframe summary is preferred, or the time duration of the video highlight when highlight summary is preferred. In addition, browsing preferences may also include descriptors describing parental control settings. A switch descriptor (set by the user) should also be included to specify whether or not the preferences can be modified without consulting the user first. This prevents inadvertent changing or updating of the preferences by the device. In addition, it is desirable that the browsing preferences are media content dependent. For example, a user may prefer 15 minute video highlight of a basketball game or may prefer to see only the 3-point shots. The same user may prefer a keyframe summary with two levels of hierarchy for home videos.

The filtering and search preferences description 532 preferably has four descriptions defined therein, depending on the particular embodiment. The keyword preferences description 540 is used to specify favorite topics that may not be captured in

the title, category, etc., information. This permits the acceptance of a query for matching entries in any of the available data fields. The content preferences description 542 is used to facilitate capturing, for instance, favorite actors, directors. The creation preferences description 544 is used to specify capturing, for instance, titles of favorite shows. The classification preferences description 546 is used to specify descriptions, for instance, a favorite program category. A switch descriptor, activated by the user, may be included to specify whether or not the preferences may be modified without consulting the user, as previously described.

The device preferences description 534 contains descriptors describing preferred audio and/or video rendering settings, such as volume, balance, bass, treble, brightness, contrast, closed captioning, AC-3, Dolby digital, which display device of several, type of display device, etc. The settings of the device relate to how the user browses and consumes the audio and/or video content. It is desirable to be able to specify the device setting preferences in a media type and content-dependent manner. For example the preferred volume settings for an action movie may be higher than a drama, or the preferred settings of bass for classical music and rock music may be different. A switch descriptor, activated by the user, may be included to specify whether or not the preferences may be modified without consulting the user, as previously described.

Referring to FIG. 26, the usage preferences description may be used in cooperation with an MPEG-7 compliant data stream and/or device. MPEG-7 descriptions are described in ISO/IEC JTC1/SC29/WG11 "MPEG-7 Media/Meta DSs (V0.2), August 1999, incorporated by reference herein. It is preferable that media content descriptions

are consistent with descriptions of preferences of users consuming the media.

Consistency can be achieved by using common descriptors in media and user preference

descriptions or by specifying a correspondence between user preferences and media

descriptors. Browsing preferences descriptions are preferably consistent with media

descriptions describing different views and summaries of the media. The content

preferences description 542 is preferably consistent with, e.g., a subset of the content

description of the media 553 specified in MPEG-7 by content description scheme. The

classification preferences description 544 is preferably consistent with, e.g., a subset of

the classification description 554 defined in MPEG-7 as classification description

scheme. The creation preferences description 546 is preferably consistent with, e.g., a

subset of the creation description 556 specified in MPEG-7 by creation description

scheme. The keyword preferences description 540 is preferably a string supporting

multiple languages and consistent with corresponding media content description schemes.

Consistency between media and user preference descriptions is depicted or shown in FIG.

26 by couple arrows in the case of content, creation, and classification preferences.

Referring to FIG. 27, the usage history description 502 preferably includes three different categories of descriptions, depending on the particular implementation.

The preferred descriptions include (a) browsing history description 560, (b) filtering and

search history description 562, and (c) device usage history description 564, as previously

described in relation to the usage preference description 500. The filtering and search

history description 562 preferably has four descriptions defined therein, depending on the

particular embodiment, namely, a keyword usage history description 566, a content usage

history description 568, a creation preferences description 570, and a classification usage history description 572, as previously described with respect to the preferences. The usage history description 502 may contain additional descriptors therein (or description if desired) that describe the time and/or time duration of information contained therein. The time refers to the duration of consuming a particular audio and/or video program. The duration of time that a particular program has been viewed provides information that may be used to determine user preferences. For example, if a user only watches a show for 5 minutes then it may not be a suitable preference for inclusion the usage preference description 500. In addition, the present inventors came to the realization that an even more accurate measure of the user's preference of a particular audio and/or video program is the time viewed in light of the total duration of the program. This accounts for the relative viewing duration of a program. For example watching 30 minutes of a 4 hour show may be of less relevance than watching 30 minutes of a 30 minute show to determine preference data for inclusion in the usage preference description 500.

Referring to FIG. 28, an exemplary example of an audio and/or video program receiver with persistent storage is illustrated. As shown, audio/video program descriptions are available from the broadcast or other source, such as a telephone line. The user preference description facilitate personalization of the browsing, filtering and search, and device settings. In this embodiment, the user preferences are stored at the user's terminal with provision for transporting it to other systems, for example via a smart card. Alternatively, the user preferences may be stored in a server and the content adaptation can be performed according to user descriptions at the server and then the

preferred content is transmitted to the user. The user may directly provide the user preferences, if desired. The user preferences and/or user history may likewise be provided to a service provider. The system may employ an application that records user's usage history in the form of usage history description, as previously defined. The usage history description is then utilized by another application, e.g., a smart agent, to automatically map usage history to user preferences.

Additional Attributes and Descriptors

In The Description and The Description Scheme

The present inventors came to the realization that additional functionality for the system may be achieved by the incorporation of particular types of information in the descriptions and description schemes. A description scheme is a data model of descriptions. It specifies the descriptors and their syntax as they are used in the description. In what follows, use the terms description and description scheme may be used interchangeably since they both correspond to describing media and user preferences. An explanation of the additional attributes and descriptors in the descriptions will be provided, followed by an example of portions of example descriptions.

After further consideration, there is a need for many users to maintain multiple separate user preference descriptions. Multiple user preference descriptions may correspond to, for example, different locations (e.g., at home, at the office, away from home, stationary versus traveling in a vehicle), different situations, different times (e.g., different days, different seasons), different emotional states of the user (e.g., happy mood

versus tired or sad), and/or persistence (e.g., temporary usage versus permanent usage). Further, the user preference descriptions may include differentiation for different terminals with different primary functionalities (e.g., a personal video recorder versus a cell phone). In addition, available communication channel bandwidth at different locations or situations may use different preferences. Also, the preference of a user for the length of an audiovisual summary of a video program for downloading may be different. The user in different usage conditions may use the user identification description scheme as a basis to distinguish between different devices and/or services. An example of different conditions may include a television broadcast receiver and a cellular telephone.

In addition to maintaining multiple user preferences for a particular user based on the aforementioned conditions, the present inventors also came to the realization that the different locations, different situations, different emotional states, different seasons, and/or different terminals (etc.), may likewise be used as the basis for distinguishing between the user preference descriptions.

One technique to permit a particular user to have multiple preference descriptions and distinguishing them from one another is by using different usernames or by using a versioning mechanism, such as a version descriptor in the identification description scheme, as described later.

As previously described, the system may include multiple user preference descriptions for a particular user. With multiple descriptions, the system may express the different user preferences with different granularity, e.g., a greater or lesser amount of

detail. The increased granularity (sparseness) may be merely the result of applying a filter to the user preference description that further reduces the amount of data. In other words, the structure of the usage preference description may be identical with the difference being the result of the filter further reducing the data. In another embodiment, the variable granularity results in a different size of the data contained in the user preferences, which may be based upon, if desired, the location and/or application of the user. User preferences with increased granularity may be especially suitable for storage on portable memory devices with limited memory capability. Likewise, the granularity may be applied to the usage history.

Another aspect of the present invention permits the user preferences (and history) to be based upon the media type, media source, or content (e.g., music versus video, radio versus television broadcast, and/or sports video versus home video). These preferences relate to the audio and/or video itself, as opposed to a third party characterization of the desirability of the multimedia. The inclusion of this information permits a reduction in the computational processing requirements depending on the media type, media source, and/or content of the media.

Another feature that may be included in the system is a protection attribute for each, or a selected set of, component of the user descriptions. The protection attributes specifies the access right of a system or service provider, typically a party other than the user himself, to the user's descriptions or any component thereof. In one embodiment, the protection attributes may be specified by a binary value that indicates the user's desire to permit others access to such data. One technique to implement the

protection attribute is to include a protection attribute as a primitive attribute that is contained by all relevant parts of the user description scheme.

Descriptors and description schemes for browsing preferences may be aligned with particular types of multimedia summary description schemes that are contained in ISO/IEC JTC1/SC29/WG11 N3246, "MPEG-7 Generic AV Description Schemes, Working Draft v2.0", Noordwijkerhout, March 2000. This allows the user to specify the type of a particular visual summary of an audiovisual program, and the duration of a summary that is in the form of a visual highlight. However, after further consideration the present inventors have determined that specification of the preferred minimum and maximum amount of data permitted in an audiovisual summary significantly enhances the system capability. Such a provision provides, for example, the capability of the user effectively browsing audiovisual summaries of content over channels with limited bandwidth and using terminals with different limitations. With a terminal connected to a bandwidth limited channel, the user may specify preference for a relatively short highlight of the program, while with a terminal that is connected to a higher bandwidth channel, the user may specify preference for a longer highlight of the program. Such a set of channels may be mobile channels and cable channels. In addition, for terminals that are not capable of displaying frames at a video rate, the user may prefer keyframe summaries consisting of a maximum number of keyframes appropriate for the communication channel bandwidth. To achieve these enhancements, the present inventors propose using descriptors in the browsing preferences description (and description scheme, or other preferences description) specifying the minimum,

maximum, and exact number of keyframes, and minimum, maximum, and exact duration of audio and/or visual highlights.

As described, the description scheme is adaptable to express the preferred minimum and maximum amount of visual material to adapt to different viewing preferences as well as terminal and communication channel bandwidth limitations. This implementation may be achieved by the following descriptors included in the browsing preferences description scheme: MaxNumOfKeyframes, MinNumOfKeyframes, NumOfKeyframes, MaxSummaryDuration, MinSummaryDuration, and SummaryDuration. The MaxNumOfKeyframes and MinNumOfKeyframes preference descriptors specify, respectively, the maximum and minimum number of keyframes in the keyframe-summary of a video program. Depending on the known bandwidth conditions of a known connection that the user uses regularly, he or she may specify these descriptors. The MaxSummaryDuration and MinSummaryDuration descriptors specify, respectively, the maximum and minimum temporal duration of an audiovisual highlight summary. Again, depending on user's taste, terminal, and channel limitations, the user may specify these descriptors. The MaxSummaryDuration and MinSummaryDuration descriptors apply to preferences for audio signals as well as where audio highlights may have been generated by audio skimming methods. User's browsing preference descriptions may be correlated with media descriptions by a filtering agent 520 in Fig. 24 in order to determine media descriptions that contain summary descriptions that match user's preference descriptions and provide the user the associated summarized media in the preferred type of summary.

An additional descriptor that may be introduced is an abstraction fidelity descriptor for universal multimedia access application, where fidelity of a summary abstraction of a program is described. This can correspond to the variation fidelity descriptor defined in ISO/IEC JTC1/SC29 WG11 N3246, "MPEG-7 Multimedia Description Schemes, Working Draft v2.0", Noordwijkerhout, March 2000. This provides an alternative to the explicit specification of the duration and bounds on the number of keyframes. A Segment Theme descriptor(s) may describe the preferred theme, or point of view, of a segment, e.g., a video or audio clip, annotated with its theme or emphasis point. For example, the theme may specify characteristics of the content of the theme. Such characterization may include a goal from your favorite team, 3-point shots from your favorite player, etc. Specifying these descriptor(s) and also ranking them enables a client application or a server to provide to the user segments according to preferred themes (and/or their ranking) matching to the their labels or descriptors at the segment level, or provide users with pre-assembled highlights composed of segments with labels matching the SegmentTheme preference.

Existing filtering and search user preference descriptions are directed to techniques of using the audiovisual content in an effective manner by finding, selecting and consuming the desired audiovisual material, while focusing on the content of the audiovisual materials. While such descriptions are beneficial, the present inventors came to the further realization that the identification of the source of the material, in contrast to merely its content, provides beneficial information for the processing and presentation of the audiovisual materials. For example, the source of the content may be from terrestrial

sources, digital video disc, cable television, analog broadcast television, digital broadcast television, analog radio broadcasts, and digital radio broadcasts. The inclusion of this information permits the user to select among these different sources and increase effectiveness by narrowing down the choices to those sources that are available to the user, such as terrestrial broadcast which is more widely available than satellite broadcast. For example, user may describe user's preference for "Star Trek" episodes that are available from terrestrial broadcast channels only.

This source distinction and identification may be performed by including a source preferences description scheme under the filtering and search preferences description scheme (or other description scheme). Accordingly, the search and preferences description scheme may include from zero or one (or more if desired) source preferences description scheme. The source preferences description scheme may be derived from the Media Format description scheme or Publication Description Scheme specified in ISO/IEC JTC1/SC29/WG11 N3247, MPEG-7 Multimedia Description Schemes, Experimentation Model (v2.0) Noordwijkerhout, March 2000.

Another feature that may be included in the system, in addition to the user's preferences, is the user's negative preferences. The negative preferences may include the user's dislikes and their relative rankings. By specifying the negative preferences, the system is less likely to select such matching preferences. This may be implemented, for example, by permitting positive and negative values to the preferencevalue descriptor.

Another feature that may be included in the system is the specification of the user's preferences as a relative preference measure of a particular set of user preferences with respect to another set of preferences, such as for example, by using BetterThan and WorseThan descriptors. This permits an implicit relative ranking of preferences even in the absence of a preference value descriptor for each preference set. This may be implemented, for example, by including Betterthan and WorseThan descriptors in the filtering and search preferences descriptions.

Expression of the Additional Attributes

The following descriptions are expressed in XML (Extensible Markup Language), incorporated by reference herein. It is to be understood that any other description language may likewise be used.

The definition of the user preference description may be as follows.

```
<UserPreference>
  <UserIdentifier protection="true" userName="paul"/>
  <UsagePreferences allowAutomaticUpdate="false">
    <BrowsingPreferences>
      ...
    </BrowsingPreferences>
    <FilteringAndSearchPreferences>
      ...
    </FilteringAndSearchPreferences>
    <DevicePreferences>
```

...

```
</DevicePreferences>
```

```
</UsageHistory>
```

...

```
</UsageHistory>
```

```
</UserPreference>
```

The primitive attributes "protection" and "allowAutomaticUpdate" may be instantiated in the UserIdentifier, Usage Preferences, and Usage History descriptions and all its relevant parts, namely, in Browsing Preferences description, Filtering and Search Preferences description, Device Preferences description, and sub-description schemes of the Usage History description Scheme.

The "allowAutomaticUpdate" attribute (set by the user) should be included in a description scheme specifying whether or not the preferences can be automatically modified (e.g., by an agent utilizing the usage history description) without consulting with the user.

The protection attribute should be included in a description specifying whether the user allows the system to make preference/history public or not. When the user agrees to make some parts of his preference/history public, for example, to service providers, the service providers can collect this information and then serve to the user contents that are tailored to the user's history/preferences. In the above example description, the user prefers to keep his username private. He also does not wish the system to automatically update his preferences.

The user identification description serves the purpose of an identifier that distinguishes a particular instantiation of the user description scheme from other instantiations for other users or other instantiations for the same user for different usage conditions and situations.

5 The username descriptor may identify a specific user from other users. In a home setting, each member of the household may be identified using a username that is unique in the household for all devices that the members of that household use on a regular basis. A username can also be used to distinguish the user description scheme of not only an individual but also a group of people, e.g., the family. Those devices that are
10 used on a temporary basis, potentially by many different people, (such as those in hotel rooms or rental cars) may assign temporary session identifications to ensure uniqueness of identifications.

 Alternatively, a version descriptor may also be included in the user identifier description to define different versions of the user descriptions (preferences and
15 usage history) associated with a particular username. Through the mechanism of the version, a person can specify different preferences and usage history, corresponding to different locations (at home, at the office, away from home, stationary versus traveling in a vehicle), different situations, different emotional states (happy versus sad), different seasons, etc. Different user descriptions are distinguished by distinct version descriptors.
20 The type of the version descriptor, may be for example, an integer, a string, or expressed as an attribute of the user identification description scheme.

The usage preference description may include a PreferenceType description, distinguishing a particular set of preferences or history according to time, or place, or a place and time combination. The definition of the usage preference description may be as shown in the following example, where place is "office" and time period is "8 hours starting from 8 AM"

```

...
<PreferenceType>
  <Place>
    <PlaceName xml:lang="en">Office</PlaceName>
  </Place>
  <Time>
    <TimePoint>
      <h>8</h>
    </TimePoint>
    <Duration>
      <No_h>8</No_h>
    </Duration>
  </Time>
</PreferenceType>
...

```

The preferencetype descriptor may be used to identify the preference type of one or more set of preferences. As previously described, a user may have different preferences depending on the user's situation, location, time, season, and so on.

The browsing preferences description may describe preferences of the user for browsing multimedia information. In essence, this description expresses the user's preferences for consuming (viewing, listening) a multimedia information. This browsing preferences description may include for example, a Summary Preferences description.

5 The browsing preferences description may include in the case of video, for example, the user's preferences for continuous playback of the entire program versus visualizing a short summary of the program. Various summary types are specified in the Summary Description Scheme in ISO/IEC JTC1/SC29 WG11 N3246, "MPEG-7 Multimedia Description Schemes, Working Draft v2.0", Noordwijkerhout, March 2000, including a
10 keyframe summary, a highlight summary, etc., where parameters of the various summary types may also be specified by summary descriptions, e.g., the time duration of the video highlight summary.

The browsing preferences description scheme may include one or more of the following non-exhaustive list of descriptors and descriptions in its description
15 scheme.

- (A) The minimum number of keyframes (MinNumOfKeyframes) and the maximum number of keyframes (MaxNumOfKeyframes) descriptors may be included. These descriptors specify the user's preference for minimum and maximum number of frames in a keyframe summary of an
20 audiovisual program. A user can specify these descriptors according to personal taste, situation, etc., and according to channel bandwidth and terminal resource limitation.

(B) The minimum duration (MinSummaryDuration) and the maximum duration (MaxSummaryDuration) descriptors may be included. These descriptors specify the user's preference for the length of a highlight summary composed of key clips in the video. These descriptors may also, for example, be applied to an audio-only material. A user can specify these descriptors according to personal taste, situation, etc., and according to channel bandwidth and terminal resource limitations.

An example for Summary Preferences description that can be included in usage preferences description is provided below.

```
</UsagePreferences>
</BrowsingPreferences>
<SummaryPreferences>
  <SummaryTypePreference>keyVideoClips</SummaryTypePreference>
  <MinSummaryDuration><m>3</m><s>20</s></MinSummaryDuration>
  <MaxSummaryDuration><m>6</m><s>40</s></MaxSummaryDuration>
</SummaryPreferences>
</BrowsingPreferences>
</UsagePreferences>
```

(C) The abstraction fidelity descriptor for universal multimedia access application relates to fidelity of a summary abstraction of a program. This preference descriptor may correspond to the variation fidelity descriptor contained in the media's variation description specified by Variation

Description Scheme in ISO/IEC JTC1/SC29 WG11 N3246, "MPEG-7 Multimedia Description Schemes, Working Draft v2.0", Noordwijkerhout, March 2000. Alternatively, the duration and number of keyframes may be defined as the fidelity descriptor.

- 5 (D) The SegmentTheme descriptor(s) may be included, which describes the theme or point of view of a segment, e.g., a video or audio clip annotated with its theme or emphasis point. An example summary preference description expressing preference for video segments (clips) labeled as "Goal from Spain" and "Replay of Goal from Spain" is as follows:

10 ...
</UsagePreferences>
</BrowsingPreferences>
<SummaryPreferences>
15 <SummaryTypePreference>KeyVideoClips</SummaryTypePreference>
<SegmentTheme>Goal from Spain</SegmentTheme>
<SegmentTheme>Replay of goal from Spain</SegmentTheme>
</SummaryPreferences>
</BrowsingPreferences>
20 </UsagePreferences>
...

- (E) The frame frequency value descriptor may be included to specify the temporal sampling frequency of video frames that can be visualized in

the browser. The frames provide a visual summary. Depending on the browser, they may also provide clickable entry points to the video. The user may click and start playing back the video starting from that frame. The frame frequency value descriptor provides similar functionality in terms of shots of the video.

The source preference description describes the preferred source of multimedia information, such as the broadcast or storage medium type (e.g., terrestrial, satellite, DVD), broadcast channel identifier, etc. An example user preference description expressing preference for Star Trek episodes available from terrestrial broadcast is as follows.

```
<UserIdentifier protection="true" userName="paul"/>
<UsagePreferences allowAutomaticUpdate="false">
  <FilteringAndSearchPreferences protection="true">
    <PreferenceValue>5</PreferenceValue>
    <CreationPreferences>
      <Title xml:lang="en" type="original">Star Trek</Title>
    </CreationPreferences>
    <SourcePreferences>
      <PublicationType>Terrestrial Broadcast</PublicationType>
    </SourcePreferences>
  </FilteringAndSearchPreferences>
</UsagePreferences>
</UserIdentifier>
```

The filtering and search preferences description includes at least one of the descriptors of preferred program title, genre, language, actor, creator of the program. An example description where user's preference is for news programs in English is given below. Such description may be included in user's smart card when he travels to Japan, for example. Note that this particular preference description is identified as being specific to Japan and differentiated by choosing an appropriate user name.

```

<UserIdentifier protection="true" userName="paul_in_Japan"/>
<UsagePreferences allowAutomaticUpdate="false">
  <FilteringAndSearchPreferences protection="true">
    <PreferenceValue>100</PreferenceValue>
    <ClassificationPreferences>
      <Language>
        <LanguageCode>en</LanguageCode>
      </Language>
      <Genre>News</Genre>
    </ClassificationPreferences>
  </FilteringAndSearchPreferences>
</UsagePreferences>
</UserIdentifier>

```

The PreferenceValue descriptor provides a technique for prioritizing filtering and search preferences, such as the value indicating the degree of user's preference or non-preference. Non-preferences may be expressed by assigning a negative (opposite) value to the preference value descriptor.

The betterthan and worsethan descriptors may describe which instantiation of preferences the user likes or dislikes relatively more compared to another instantiation, where different instantiations are identified using the filtering and search preference type descriptor. This provides robustness against changes in the preference value descriptor automatically, for example, by an agent.

The filtering and search preferences description may also contain a description of a preferred review to express user's desire for searching for programs that are favorably reviewed by specific individuals. For example, preference for movies reviewed by movie critics Siskel and Ebert and found to be "two-thumbs- up" may be described and included in the filtering and search preferences description. An overview of the entire description scheme is shown in FIG. 29. With the ever increasing amount of available media, such as audio, image, and videos, it becomes increasingly more difficult for a user to select desirable media for subsequent consumption. The user may manually peruse program listings to select the desired material. However, the manual selection of media from an exhaustive program listing is time consuming and inefficient.

As previously discussed, a description scheme, such as those for the user, programs, and system, provides a structure within which is captured information regarding (1) the user's preferences, (2) the system, and (3) the programs themselves. By processing the information contained within the user's usage description scheme and the program description scheme of available programs, the system may determine those programs that are most likely desirable to the particular user. The processing by the system for such information may be referred to as an agent.

Existing agents are focused on correlating a limited number of user preference descriptors with a limited number of program descriptors. The designer of such agents manually determines, and hard codes into the agent, predetermined interrelationships which are likely to result in identifying desired programs. As such, the mapping between the user preference descriptors and the program descriptors includes a static model because such designers are under the belief that the domain of data fields is a fixed predetermined set, and therefore the relationships between the potential combinations of relevant data is likewise a fixed predetermined set. For example, the "actor" in the user preference and the "actor" in the program descriptor may be a relevant potential combination. The traditional focus for designing such static agents avoids the problematical dilemma of how to interpret and process an arbitrarily complex set of preferences.

Maintaining the traditional focus of avoiding an arbitrarily complex set of user preferences, commercial products such as TiVO and Replay TV, permit the specification of a first preference, such as a particular actor. The user may further attempt a more specific search by searching for a first preference, a second preference, and additional preferences. While this results in identifying the desired programs, it is a time consuming and frustrating process for the user. Like the static agents, the TiVO and Replay TV devices have a limited set of permitted search queries.

While such static models of the interrelationships is readily easy to implement, it results in a system that is unable to process interrelationships that are not foreseen by the agent designer. The present inventors came to the realization that all of

the potentially desirable interrelationships, especially for an arbitrarily complex set of preference criteria, can not be effectively programmed using the traditional static model.

Referring to FIG. 30, a filter agent 600 receives or otherwise has access to at least one program description 602 and at least one user preference description 604.

5 Each program description 602 may also, if desired, include meta-data associated with the actual consumable program media. Also, the user preference description 604 contains selection criteria for the information contained within the meta-data. The output of the filter agent 600 is a subset of the received program descriptions 606 that have been selected, and tailored if needed, in accordance with the user preference description 604.

10 Referring to FIG. 31, the filter agent 600 receives the user preference description 604 and interprets the information contained within the user preference description 604 at block 610 using information from a mapping table 614. The filter agent 600 also builds a model , such as a combinatorial model, of the user criteria indicating the desired user criteria at block 612 using information from the mapping table
15 614. The resulting model or otherwise set of criteria, is then applied against the available program descriptions 602 at block 616 to select the desired programs. Each of the selected programs include a rich set of associated data which may then be applied against user criteria at block 618 to further refine the data by extracting desirable sub-portions of each program. Each of the selected programs may further be cloned at block 620 together
20 with the desired sub-portion of each program, and the resulting tailored instances are output from the filter agent 600.

Referring to FIG. 32, a Program Description may be composed of a hierarchy of individual descriptions. The hierarchy implies relationships amongst the individual description elements including composition, type-of, and other relationships. The particular structure of the input Program Descriptions vary, and are typically generated and provided by a commercial provider. The output Program Descriptions may be, for example, copies of the selected input instances, portions of the selected input instances, or are modified clones of the input instances. In the case of modified clones, the clones should describe a subset of the program media that meets the user's preferences, and exclude the portion that the user does not desire, or is not feasible to process for other reasons, such as bandwidth. It is to be understood that the output may omit cloning, if desired.

Referring FIG. 33, the User Preference Description may include primitive elements that describe individual or multiple preferences. The individual preferences may be generally defined in terms of "name : value" pairs. The name component is one of a set of Preference Names. The domain of the value depends on the name, such as codes and free form text.

The individual preference may be a single preference test (or multiple preference). It is to be understood that the individual preferences are not limited to tests. For example, the User Preferences may describe the desired configuration for presentation, such as volume or any other functionality. Each preference test describes some aspect or attribute of the Program Description that is to be evaluated. If desired, the preference test may be limited to the smallest granularity of a test that may be executed

on the Program Description. A common preference test is a comparison of a Program Description element's value to the preference value. It is also to be understood that the preference tests need not be directly associated with the value of a corresponding element, if any, of the Program Description. For example, a single User Preference element, indicating a desired number of key-frames to be shown, may be tested against the count of elements in the Program Description representing (or describing) a key-frame. In general, the pair (title: title_value) will compare the title element value of the Program Description to title_value.

After further consideration, the present inventors came to the realization that the individual preferences may include composite relationships. Moreover, the relationships may include duplicate fields, such as several instances of "name" in either the user preferences and/or the Program Descriptions. With the inclusion of composite relationships it becomes difficult to determine an appropriate technique for queries, where duplicate individual preferences are at the same or different hierarchy levels. In addition, it is difficult to determine how to interpret queries that provide multiple matching results (such as several instances of "John Doe") or inconsistent matching entries (such as several instances of "John Doe" and a lack of an instance of "Comedy"). For example, referring to FIG. 34, if the user uses a query involving multiple preference names, and the query results in several potential matches, it is difficult to determine if an appropriate program has been located or which portion of an appropriate program is suitable. As shown in FIG. 34, the preference with name A is composed of one instance of name B and two instances of name C, each of which may include the same or different names.

Referring to FIG. 35 an example of a portion of a User Preference Description is illustrated. This portion of a user preference description illustrates a hierarchy of a "creator" that has the "role" of "author" with the name of "Thomas" "Aquinas". In addition, the hierarchy may be used as the path to define the interrelationships.

The normal behavior of a location path is to retrieve the single data from this node in the program. One potential enhancement is to allow this data to be manipulated and combined with other related nodes to form a composite value.

One example is when evaluating a media review rating, three numerical values may be provided, namely, RatingValue, WorstRating, and BestRating. A composite value for media review rating may be calculated as $((\text{RatingValue}) - (\text{WorstRating})) / ((\text{BestRating}) - (\text{WorstRating}))$.

Another example may include the test of Keyword preferences against the Title or Description fields by concatenating these two fields. A composite value might be calculated as $(\text{CreationDescription}/\text{TextAnnotation}) \& (\text{Title}/\text{TitleText})$. It is noted that these two fields use relative paths from the parent "Creation" element.

Yet another example may include a single preference data manipulated to adjust its value numerically, or its text may be translated into a target language.

The composite values provide defaults for any of the calculated elements. This is useful for defining the default range of a media review rating. It is also useful for inserting identity values (e.g. 1, 0, "") when the absence of an element should not make the test fail.

The Preference Description may make use of built-in composite values. An example of built-in composite value may be based on the environment of the viewer. For instance, a portion of a Preference Description may define time of day ranges when the user wants the associated preferences to be evaluated. The target location could be defined as a composite value of built-in type "TimeOfDay".

Referring to FIG. 36, the present inventors have determined that a mapping table of the User Preferences and the input Program Descriptions provides a robust comparison of the user preferences and the input Program Descriptions. It is to be understood that the mapping table may be any type of data structure, as desired. The first column ("name") of the mapping table includes the name of one or more of the user preferences. FIG. 36 illustrates the inclusion of the user preferences of FIG. 35. Each node, generally referred to by name, of an individual preference to be tested has an ancestry path associated with it. The second column ("location") of the mapping table includes the name of one or more of the input Program Descriptions. Alternatively, portions of the path or even single elements of the path may be specified in the table, if desired. For example, the Creator/Individual/FamilyName preference in FIG. 35 has a path of /FilteringAndSearchPreferences/CreationPreferences/Creator/Individual/FamilyName. This path is decomposed and resolved, piece by piece, using the "Location" column (e.g., field) in the Mapping Table. The result of decomposing and resolving the user preference path results in a corresponding path within the Program Description resolved from entries in column two. For example, the

resulting location path for this test may be "/Program/CreationMetaInformation/Creation/Creator/Individual/FamilyName".

Common names, such as "Country" used at multiple locations, may be distinguished by including all or part of the ancestry path. For example, the following two preference tests have the same "leaf" name, but it may be desirable to have different tests for each. This may be done by specifying more of the ancestry in the Name field (column 1) of the mapping table: "/FilteringAndSearchPreferences/CreationPreferences/CreationLocation/Country", and "/FilteringAndSearchPreferences/ClassificationPreferences/Country". To distinguish between the two, the following names may be used: "/CreationLocation/Country" and "/ClassificationPreferences/Country". In addition the preference tests may be associated with multiple entries in the Mapping Table. This permits a single test to be performed on more than one location in the Program Description.

The Location field may include various wildcards to expand or restrict the target paths to be evaluated in the Program Description. For example, a "*" wildcard implies that there may be multiple instances of the given location under one parent, e.g., /Creation/*Creator implies that there may be multiple Creators under the Creation parent. A "#xxx" wildcard restricts the target path to the xxx instance of the given location under its parent, e.g., /Creation/#002Creator restricts the target path to the second instance of Creator under Creation. A double forward slash "/" indicates a node of the target path which may be used as a base path for groups of tests which must be constrained to evaluate from the same common location. In particular, this is useful for Constrained-

AND operations, described later. The preference paths may be used to build target locations in the program. These preference paths may also allow preference paths to be interpreted as locations. Composite values may be defined for these preference path locations.

5 Syntax for a default preference and a default location may be provided. This allows updates in the preference or program definition to be handled by the filter agent without requiring changes to the mapping table.

10 The default mapping elements may be specified for a limited set of preference branches to bound the default mapping to a safe portion of the user preferences.

15 For instance, the default element "FilteringAndSearchPreferences/CreationPreferences/UserDefinedPreference/*" may place a default mapping that can only map to elements in the program beneath the "Program/CreationMetaInformation/Creation" branch.

20 The third column "TestOp" of the Mapping Table includes what comparison to perform between the corresponding user preference path (column 1) and (resolved) input Program Description location (column 2). In this manner, the Mapping Table provides a convenient manner of identifying the interrelationships between the corresponding data from the user preferences and input Program Descriptions. For instance, the "FamilyName" preference in FIG. 35 has a test operator of substring-case-insensitive when compared with "/*FamilyName". Test operators may yield a discrete result, such as true or false, they may yield a range of values, or any other desired data.

In particular results that span a range of values provide the advantage that filtered programs may be sorted according to the resultant "similarity" value. This provides the user with a ranked output that they may select from. Also, user preferences may be "softened" to pass programs that are near matches to the specific preference criteria. This fuzzy approach may allow the user preference description to more clearly model the user's intended criteria. In cases where the entry is always a parent (composed of children preference tests) the test operator may be NA (not applicable). An exemplary set of test operators are illustrated in FIG. 37.

After the individual preferences are interpreted into individual preference tests, these tests may be combined into a single test that models the user's preferences. The preferred technique includes combining the individual preference tests according to their hierarchy. Each parent test becomes the combination of its children tests, and this continues up to the root preference, yielding in effect one composite test. The combination of "children" tests within a single "parent" may be broken down into the combination of similar tests and the combination of dissimilar tests. Similar tests may have the same name or otherwise be associated in the Mapping Table such as by being on the same row. Also, dissimilar tests may have different entries in the Mapping Table.

It is to be understood that the concept of inter group and intra group interrelations relates to any comparison between different sets of data, whether or not they include a hierarchical scheme. As an example, intragroup may be used to define a group of similar tests. Also, any scheme may be implemented to form comparisons or groupings for the testing of data.

If desired, the mapping table, which may be any type of data structure or otherwise to simply express the desired operations to be performed, may be expanded to include additional functionality. For example, specific groupings of user preference may be denoted, to specify additional operations to be performed on the elements of the group that are separate from the inter group and intra group operations. These specific groupings may provide additional flexibility for combining individual preference tests. The combinatorial operations applied to these groups may be performed before, after or instead of the general inter group and intra group combinatorial operations.

For instance, entries in the mapping table may be explicitly linked together with a shared index, and a specific combinatorial operator may be mapped to each indexed group. The UserPreferences elements may likewise be explicitly linked together with a shared index. The latter two groups and operators present an alternative method to generate the arbitrarily complex combinations, without using one of the four methods for generating all typical permutations, shown in FIG. 13. A preferred sequence for performing the various combinatorial operations might be intra group operation, followed by indexed group operation, followed by inter group operation.

In addition to explicitly defined indexed groups, other groupings may be built-in. For instance, a program description may have attributes associated with it. The user preferences that are mapped to this program description and its associated attributes may be grouped together in a so-called attribute group, and a specific combinatorial operator may be mapped to this attribute group. For example, the program description element, TitleText, may have a language attribute associated with it. A user preference,

KeywordPreferences, may be mapped to TitleText and a separate user preference may be mapped to the language attribute of TitleText. These two user preferences may be grouped together into the following attribute group, and the results to these two tests may be combined in an attribute group combinatorial operation:

5	Name	Location	AttrGroupOperation
	KeywordPreferences+xml.lang	Title/TitleText+xml.lang	AND
	KeywordPreferecnes	Title/TitleText	AND

The functionality may also include multi-mapped preference group and associated operator. Elements in this group may have the same user preference element, but have multiple different program description mappings. For example, PersonName may have the following mappings, forming one multi-mapped group:

10	Name	Location	MultiMapGroupOperation
	PersonName	Creator/GivenName	OR
15	PersonName	Creator/FamilyName	OR
	PersonName	Creator/ProfessionalName	OR

Preferably, the various groupings are combined in sequence starting with attribute groups, followed by intra groups, multi-mapped groups, indexed groups, and inter groups.

20 Referring to FIG. 38, exemplary examples of combinatorial operators are listed. Several of the combinatorial operators (notably SAND, SUM, FREQ, and RATIO) provide "soft" combinations that can be used to yield non-zero results, even when some of the individual preference tests in the combination are zero. These soft combinations are useful when a group of program descriptions are to be evaluated, but

one or more nodes in the group does not exist. In this manner, the result will be a diminished, but non-zero value.

For example, the SAND operator provides a soft AND combination of its constituent elements by applying a transformation to the input values before they are combined. This may transform a zero input to a non-zero value. Additionally, the combination operation may be a non-linear function that will increase or decrease the result, related to a strict AND combination.

Another set of combinatorial operators are soft maximum and soft minimum operators. In the typical maximum or minimum operation, only one of the combined individual preference tests determines the combined result value. In contrast, the soft minimum operator and soft maximum operator allows other non-contributing individual preference test results to adjust the final combined result. Typically, the adjustment is a minor amount, e.g., +/- 10 percent. The purpose of the soft maximum/minimum operators is shown in the example where a user prefers program which contain A or B. IF a program with A and a program with A and B were available, the typical maximum operator would rank both programs equally, whereas the soft maximum operator would rank the program containing A and B above the program containing only A. A similar result occurs from the soft minimum.

Another combinatorial operator is an average, which averages a set of scores resulting from a plurality of tests.

One combination for dissimilar preference tests is under a single parent. Each entry in the Mapping table has a field that defines how this type of preference test

should be combined with different type preference tests under the same parent. This type of test may be referred to as inter group combinatorial operator (InterOperator).

Referring to FIG. 39, this example illustrates a parent test with four dissimilar "leaf" test children. Two of the leaf tests have InterOperator AND. These two tests are evaluated independently, then their results are AND'd together. The other two leaf tests have InterOperator OR. These two tests are evaluated independently, then their results and the AND'd sub-result are all OR'd together to form the parent test result.

The rules for combining dissimilar tests (with the operator mappings of OR and AND) may be:

- (1) evaluate all the tests;
- (2) AND the test results which have InterOp=AND, forming the InterAND result;
- (3) OR the test results which have InterOp=OR, with the InterAND result, forming the InterOR result; and
- (4) the InterOR result is the final result for the parent test.

In many cases, preference tests of the same type under a single parent will have a specific desired combination for those preferences before they are combined with the other different children of that parent. Each entry in the Mapping Table has a field that defines how this type of preference test should be combined with similar type preference tests under the same parent. This may be referred to as the intra group combinatorial operator (IntraOperator). Referring to FIG. 40, the parent test has four dissimilar children and four pairs of similar children. The similar children are first

combined into intra groups using either respective IntraOperator. Then the intra group results are combined with the other groups and tests using their respective InterOperator.

The rules for combining similar and dissimilar tests (with the operator mappings of OR and AND) may be, for example:

- (1) evaluate all the tests;
- (2) group together test results that have the same name, forming Intragroups;
- (3) AND the test results within Intragroups which have IntraOp=AND, forming IntraAND results;
- (4) OR the test results within Intragroups which have IntraOp=OR, forming IntraOR results;
- (5) AND all the solitary test results (not part of an Intragroup) and Intragroup results which have InterOp=AND, forming the InterAND result;
- (6) OR all the solitary test results and Intragroup results which have InterOp=OR, with the InterAND result, forming the InterOR result; and
- (7) the InterOR result is the final result for the parent test.

The general case of intra group combinations shown in FIG. 40 has the special default case shown in FIG. 41. This simplified approach supports the most common intra group operation, OR'ing and the most common inter group operation, AND'ing. One of the advantages of the approach of FIG. 40 is that the most common operations are supported with reduced constructs, and other combinations are supported by duplicating the hierarchy as necessary. The default mapping allows field updates to

the User Preferences Description schema without requiring a change to the application or Mapping Table.

An example of a default mapping may include defining a parent (e.g., node) in the user preference that maps to a parent (e.g., node) in the Program Description and setting a "default" comparison between the two. In the event that an additional child user preference is added to the parent in the hierarchal tree then this child is automatically mapped to a corresponding child of the parent in the hierarchal tree of the Program Description. Preferably the two children have the same name to simplify identification.

The example illustrated in FIG. 42 show the four permutations for combining two leaf tests that may be of type A or B, namely, AND dissimilar types, OR similar types, OR dissimilar types, and AND similar types.. In addition, these leaf tests could also be parent tests themselves, composed of their own sub-hierarchy. The simplified approach relies on there being a layer (below the topmost layer) that has IntraOperator AND, while the rest of the hierarchy further down uses IntraOperator OR. This supports combining similar or dissimilar tests by merely going up in the hierarchy a sufficient number of levels.

The location mappings described in the Mapping Table yield global paths that start from the root node in the Program Description ("/Program"). Some preference tests may require support for a relative path. A special form of the InterOperator AND is defined which constrains a group of tests to be performed on the same element or parent element in the Program Description. This is defined as a Constrained-AND (CAND) combinatorial operator.

The constrained operation has a base path and multiple tests. The base path defines the starting node for all the predicate tests to be performed from. In the general example illustrated in FIG. 43, the parent of the three Constrained-AND tests is "P". The mapped location for "P" in the Program Description is the base path, which resolves to "r/s/t/p". Therefore, for each instance of "r/s/t/p" in the Program Description, the children elements "x" and "y/z" are tested by "X", and the two "Y" tests.

A user trying to find programs on stuffed pasta might create the following profile fragment:

```
<KeywordPreferences xml:lang=en> calzone </KeywordPreferences>
<KeywordPreferences xml:lang=en> tortellini </KeywordPreferences>
<KeywordPreferences xml:lang=en> ravioli </KeywordPreferences>
```

The word calzone is a type of stuffed pasta in English, but it is underwear in Spanish.

Without the use of Constrained-AND, the agent may erroneously retrieve programs such as

```
<Creation>
  <CreationDescription xml:lang=en>Victoria Secrets models women's underwear</CreationDescription>
  <CreationDescription xml:lang=es>El Secreto de Victoria se demuestra los calzones de mujer</CreationDescription>
</Creation>
```

The example shown in FIG. 44 depicts the Constrained-AND operator used for the Creator preference. A special syntax in the Mapping Table indicates that the base path is "/Program/CreationMetaInformation/Creation//Creator". Therefore, the predicate tests are performed against the node instances of "Creator". The first Program Description examples passes this Constrained-AND test, while the second fails. Notice

that the second Program Description would have passed a test that was composed of regular AND operators and their global paths.

As shown in FIG. 44, if all the CAND's were all regular AND's, then the User Preference would be asking:

(1) Are there any nodes of path

"/Program/CreationMetaInformation/Creation/Creator/
role" that have value matching "Author" AND;

(2) are there any nodes of path

"/Program/CreationMetaInformation/Creation/Creator/
Individual/FamilyName" that have value matching
"martin" AND;

(3) are there any nodes of path

"/Program/CreationMetaInformation/Creation/Creator/
Individual/FamilyName" that have value matching "Martin".

This test would pass both Program Descriptions shown in the example.

As shown in FIG. 44, with the defined CAND's, the User Preference is asking:

(1) are there any nodes of path

"/Program/CreationMetaInformation/Creation/Creator", then, at each instance of
these nodes;

(2) are there any child nodes of path "role" that have value matching "Author"
AND;

(3) are there any child nodes of path "Individual/FamilyName" that have value matching "Martin" AND;

(4) are there any child nodes of path "individual/GivenName" that have value matching "Steve".

5 This test would only pass the first Program Description. This illustrates that the user of AND and CAND operators on the same program description may result in different results.

Referring to FIG. 45, a general example demonstrates Inter Operators, Intra Operators, and Constrained Operators. The rules for combining the tests may be, for example:

10 (1) group together tests which have InterOp=CAND, forming the CAND group;

(2) determine the base path for the CAND group from the lowest common Program Description path indicated in the Mapping Table;

15 (3) for each path instance in the Program Description that is equivalent to this base path, evaluate all the tests within the CAND group, from this path instance;

(4) evaluate all the tests not within the CAND, from the root path;

(5) group together test results within Intragroups which have IntraOp=AND, forming IntraAND results;

20 (6) AND the test results within Intragroups which have IntraOp=AND, forming IntraAND results;

(7) OR the test results within Intragroups which have IntraOp=OR, forming IntraOR results;

(8) AND the solitary test results (not part of an Intragroup) and Intragroup results which have InterOp=CAND, forming the InterCAND result;

5 (9) AND the solitary test results and Intragroup results which have InterOp=AND, with the InterCAND result, forming the InterAND result;

(10) OR the solitary test results and Intragroup results which have InterOp=OR, with the InterAND result, forming the InterOR result; and

(11) The InterOR result is the result for the parent test.

10 An illustrative example of one embodiment of the technique described herein includes the example illustrated in FIG. 46 together with the resulting Mapping Table illustrated in FIGS. 47A-47D. It is noted that the default for InterOp/IntraOp operations are AND/OR. It is also noted that the preferences just below the highest level (CreationPreferences, ClassificationPreferences, SummaryPreferences) are AND/AND.
15 Also some of the composite preferences such as Creator have child preferences that are CAND/OR. Further, the multiple Filtering And Search Preferences may be distinguished by Preference Type attributes. The IntraOp for multiple Filtering And Search Preferences and multiple Browsing Preferences is specified.

20 The multiple User Preference elements may contain a ranking attribute. Such ranking attributes may be applied at each comparison test and each combinatorial operation, to yield a composite ranking score. this may be utilized to provide a sorted test result for the user.

Referring to FIG. 48, the user preference hierarchy of (name : value) pairs may be supplemented with attributes regarding the intended combination of individual preference tests. As shown in FIG. 19, the supplementation may indicate that the two tests of type A should be AND'd together and this result OR'd with the other tests under this parent. The primary advantage of this supplemental enhancement is that the user preference description may override the default behavior for specific tests. This makes interpretation of the user preference description more flexible so that it may be tailored to the user's specific preference criteria.

The discrete implementation of the filter agent will yield as output a group of program descriptions that are merely members of the input set. The output group may actually just be a list of the input Program Descriptions that passed the selections. However, there can be components of the User Preference Descriptions that are well suited to extract a subset of the whole Program E... an output more tailored to the user's preference criteria. For ins... maximum number of key frames in order to prevent overlo... es of their system.

The process of cloning the selected input Program Descriptions and modifying them to include a particular desired subset by the user may achieve enhanced benefits. The modified Program Description is a clone of the input because it refers to the same base set of Program Media. However, it is modified to refer to the subset of the Program Media that is desired by the particular user. In some cases this may result in

smaller quantity of the program being available. In other cases, this may result in different summaries of the program, though it refers to the full program.

The cloned Program Description provides a more succinct representation of what the user prefers. In this manner, it may not be necessary to annotate or provide additional identifiers to describe what the user actually desires.

In a modular implementation, the filter agent may not be closely coupled with the media manager and the presentation processes. In this case, the cloned Program Description offers a standardized format for describing the desired program, without having to create a new syntax or an application programming interface (API).

The cloned Program Description may also be used to create a "pull" for Program Media that will yield only the desired portions of the media. This provides a convenient technique for a media provider to provide to the user only that specific media that is desired.

A service provider may likewise provide service to the user according to the user's preference where service includes a modified cloned program description. The cloned description may be a subset of the complete "rich" program description that is usually maintained by the service provider. The clone may contain varying levels of "richness". This permits the provider to offer various service levels to its clients.

The cloned Program Description also allows the customer and/or service provider to tailor the amount of material that will be transmitted to the customer. This enables the quantity of material to be matched to the available memory in the client device and the available bandwidth of the delivery channel.

The cloned program descriptions may provide a memory efficient way of storing descriptions of selected programs in the client's local storage.

One technique to achieve cloning is cloning by "addition", as illustrated in FIG. 49. The core elements of a Program Description are identified and copied into the clone. These items typically include the root "/Program" element, the
5 "/MediaInformation", etc. To this core set, the extractor adds the desired components for the user. Also, some adjustment may be necessary to resolve interdependencies of the extracted elements. For example, the Program Description may contain elements or groups of elements that are extensions or refinements of other elements in the Program
10 Description. These extension elements may refer to the base elements without actually duplicating the base elements. In this instance, if the extractor should extract the extension elements but not the base elements then all the base elements must be inserted into the cloned Program Description to make it accurate.

Another technique to achieve cloning is cloning by "deletion", as
15 illustrated in FIG. 50. The entire input Program Description is cloned for the output. The extractor then builds a list of the desired components that are to be retained in the output. Then, optional elements are identified in the clone. If these optional elements are not included in the list of elements to be retained, then they are deleted. Thereafter, some adjustments may be necessary to resolve interdependencies of the extracted elements, as
20 previously described. One advantage of cloning by deletion over cloning by addition is that it is less susceptible to changes in the Program Description. Items that are not explicitly listed as being optional will be passed on. In effect, this method errs on the

